Charith Perera (Eds.)
PhD, MBA

Internet of Things

Systems Design

Project Workbook

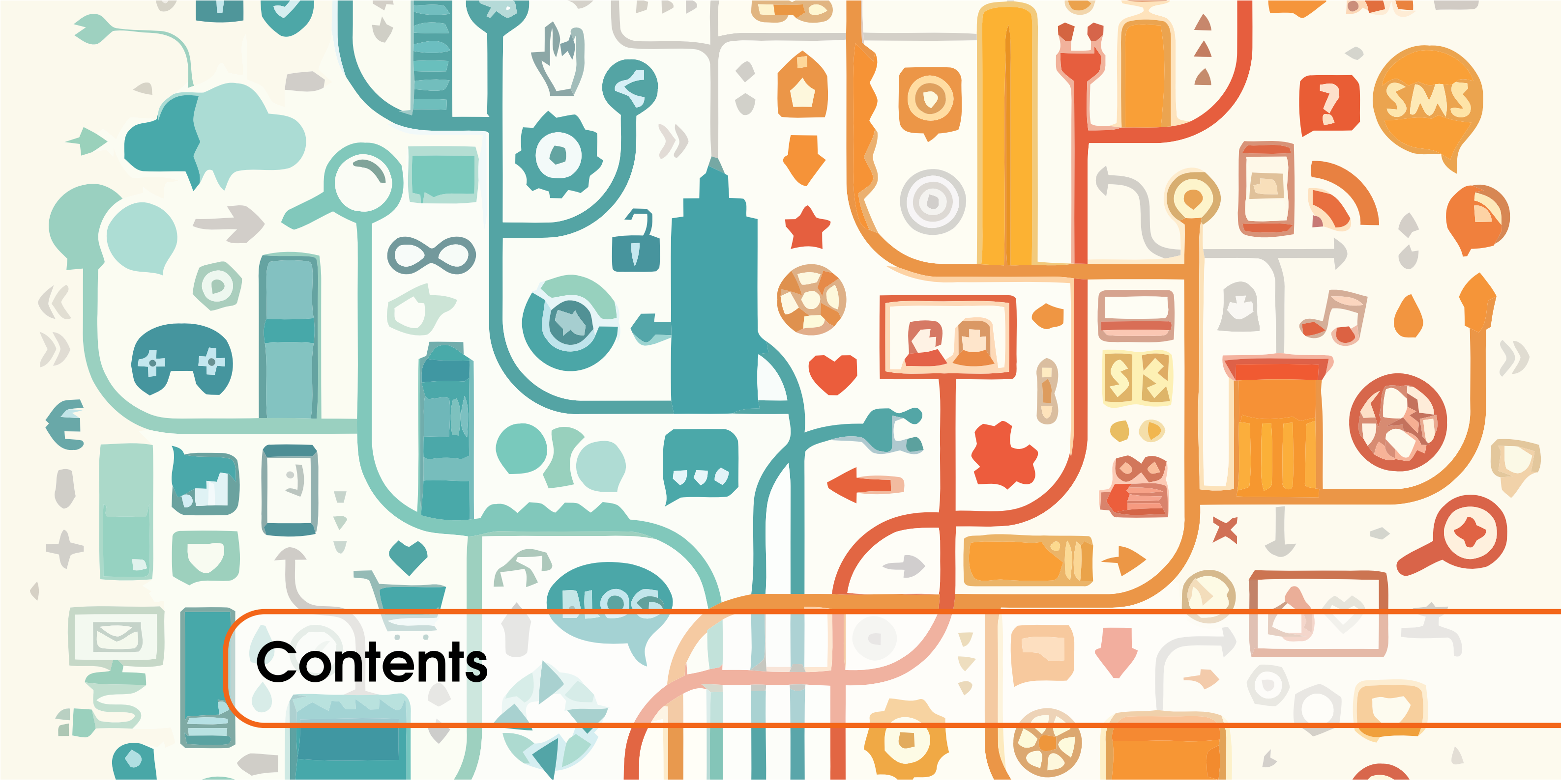
Published by IOT Garage

Brand names and trademarks remain property of their respective owners.

How to cite this book:

Charith Perera, Internet of Things: Systems Design Project Workbook, IOT Garage, 2025

Version 1.0 (Latest), February 2025



Contents

Preface

This IOT PROJECT WORKBOOK is primarily compiled to support the university courses on *'Internet of Things: Systems Design'* at both undergraduate and postgraduate levels.

Welcome to the IoT Systems Design and Development Workbook. The aim of this workbook is to assist learners in organizing and developing complex IoT systems methodically. While the activities presented are optional, they are designed to help teams and individuals streamline their processes, clarify their ideas, and debug issues more effectively.

This workbook aligns with the structure of our module, assuming familiarity with topics such as applications and use cases, architecture, sensing and actuation, networking and communication, data management and analytics, privacy and security, human factors and interaction, and design strategies and prototyping. Our workshops are divided into five different focus areas, each integrating knowledge from these eight lessons.

At the beginning of each workshop, we will highlight the key lessons you should have completed and the aspects to consider during the workshop. Although we have compiled a cohesive series of workshops from various existing materials, these materials were developed independently. As a result, transitions between workshops may feel less smooth. This is intentional, as we aim to expose you to a broad spectrum of tools and techniques available in the IoT field, enhancing your long-term ability to make informed choices tailored to your needs.

We would like to extend our gratitude to all the individuals who developed the tools and materials incorporated into these workshops. Their contributions have enabled us to create a rich, interactive, and engaging learning experience that will aid you in systematically designing and developing your IoT systems.

It is important to note that as learners, you might discover more effective ways to use the techniques and materials we introduce. While we present one approach to utilizing these resources, it is not the only way. We encourage creativity and innovation, inviting you to explore and develop your own methods. Although we recommend trying the various materials, tools, and techniques provided, the exposure to different options is key to your growth and learning.

Thank you for embarking on this journey with us. We hope this workbook will be a valuable resource in your IoT projects, helping you navigate the complexities of design and development with confidence.

Workshops Overview

This section provides an in-depth exploration of five carefully designed workshops, each focusing on a distinct aspect of IoT system design and development. These workshops are intended to build your knowledge and skills incrementally, equipping you with the tools and understanding necessary to tackle the multifaceted challenges of IoT systems. Each workshop has a clear primary objective supported by secondary objectives to ensure a comprehensive learning experience. Together, these workshops encourage iterative learning, critical thinking, and the ability to integrate diverse concepts into cohesive solutions.

Workshop 1: Getting Started with the Internet of Things

The primary objective of this workshop is to integrate foundational concepts such as sensing, actuation, and data management to conceptualize and develop innovative IoT systems. Secondary objectives include exploring the selection and application of sensors and actuators, as well as understanding how to manage and analyze the data collected from these components. Participants will engage in brainstorming exercises to identify suitable sensors and actuators for their chosen IoT applications, considering the implications of these choices. Key questions such as "What data is essential for your application?" and "How will you process this data?" will guide the exercises, fostering a deeper understanding of the complexities involved in IoT system development. This foundational workshop lays the groundwork for subsequent sessions by introducing the core elements of IoT design.

Workshop 2: Planning and Architecting Systems

The core objective of this workshop is to build on the ideas generated in Workshop 1 by focusing on system architecture, networking, and communication. Secondary objectives include evaluating architectural trade-offs, exploring networking protocols, and understanding how to optimize communication within IoT systems. Participants will take their initial conceptual designs and refine them, diving deeper into the technical aspects of architecture and networking. They will consider questions such as "Which architectural patterns best suit your application?" and "What network protocols are optimal for your chosen environment?" Through this process, participants will gain a practical understanding of the decisions required to develop robust, scalable IoT systems. This workshop emphasizes the importance of coherence between architectural and networking elements to ensure system efficiency and reliability.

Workshop 3: Internet of Things Product Development

The primary objective of this workshop is to focus on human factors and interaction design while advancing the prototyping of IoT solutions. Secondary objectives include revisiting and refining designs from previous workshops to enhance usability and stakeholder engagement. Participants will consider the user experience by addressing questions such as "How will your system interact with end-users?" and "What features make your design practical and appealing?" Additionally, this workshop will emphasize the importance of iterative prototyping, helping participants prioritize key aspects of their designs. Discussions will also cover the aesthetic and functional aspects of IoT products, encouraging participants to explore creative approaches within resource and time constraints. By the end of this workshop, participants will have developed a user-centric IoT solution ready for further refinement.

Workshop 4: Introduction to Privacy by Design Schemes

The core objective of this workshop is to ensure privacy-preserving measures are embedded in IoT designs. Secondary objectives include identifying privacy risks associated with IoT solutions and exploring strategies to mitigate these risks. Participants will revisit their earlier designs to incorporate privacy-by-design principles, addressing questions such as "What are the potential privacy vulnerabilities in your system?" and "How can these vulnerabilities be mitigated?" This workshop highlights the ethical and legal responsibilities of IoT developers, emphasizing the importance of protecting user data. Through discussions and exercises, participants will gain an understanding of how to align their designs with privacy requirements, making their solutions both secure and trustworthy.

Workshop 5: Non-Functional Requirements for Internet of Things

The primary objective of this workshop is to identify and address non-functional requirements critical to IoT system performance. Secondary objectives include balancing functional and non-functional requirements, navigating trade-offs, and iteratively refining designs to meet operational criteria. Participants will explore aspects such as scalability, reliability, and security, understanding how these factors influence overall system performance. They will engage with questions like "What trade-offs must be made to balance competing requirements?" and "How do non-functional requirements shape the deployment of your system?" This workshop challenges participants to think beyond immediate functionality, fostering a holistic approach to IoT design that integrates both technical and contextual considerations. By iteratively revising their designs, participants will ensure their IoT solutions are robust, efficient, and aligned with deployment objectives.

Optional Participation and Non-Graded Deliverables

This workshop (and any associated activities or outputs) is entirely optional and carries no official grading. Completing the workshop or producing any of its suggested deliverables will **not** affect your overall course assessment in any way. These exercises and artifacts serve solely as a platform for collaborative learning, exploration, and discussion.

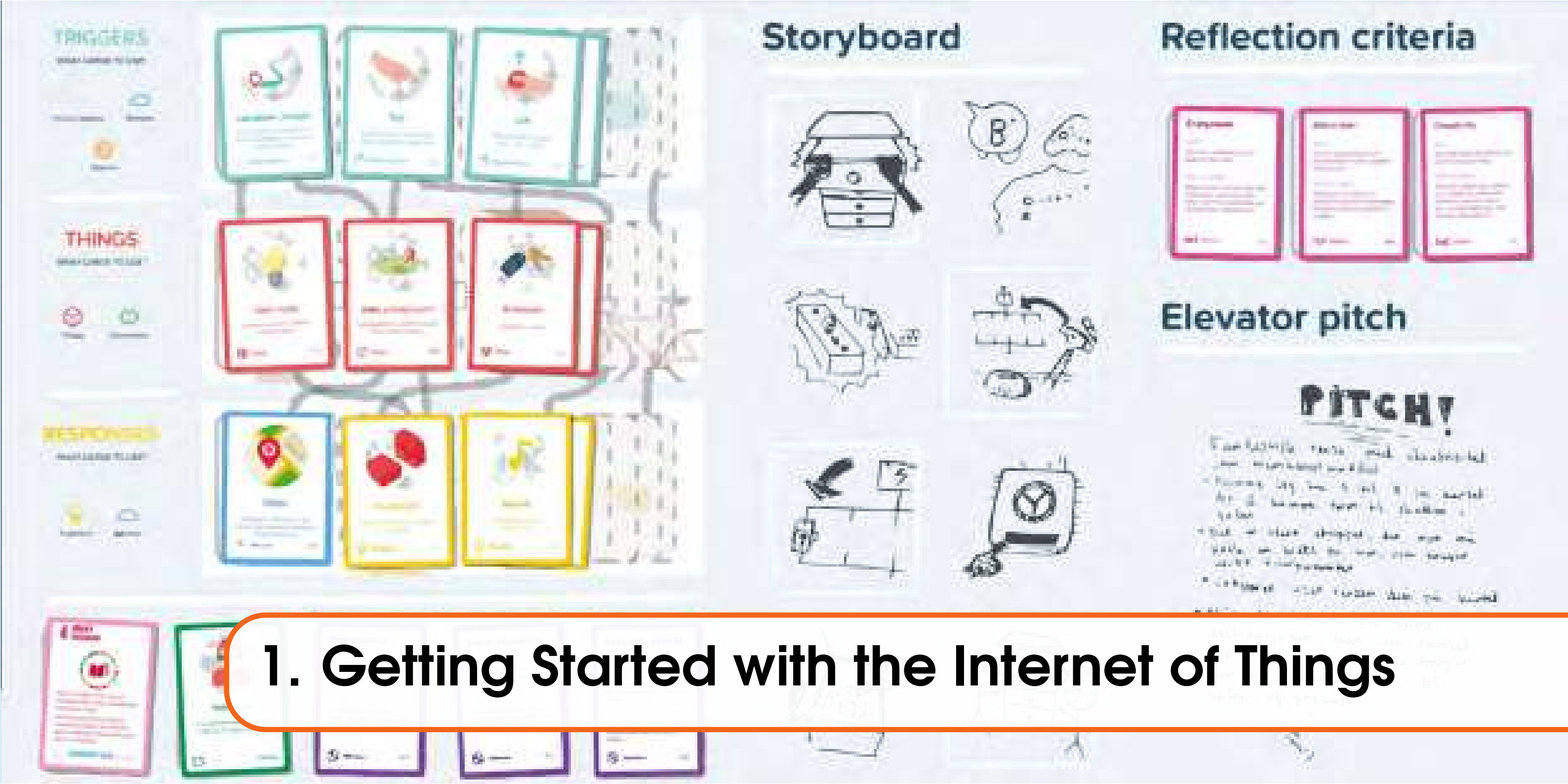
Accessing the Workshop Material

You can follow these workshops using either digital or physical materials.

- **[Official]**: You can download the workshop materials from the official Git repository (◆). The repository contains all the original materials created by the respective owners.

  `https://gitlab.com/IOTGarage/iot-project-workbook`

- **[Unofficial]**: Alternatively, you can use the digital materials we have organized as Miro boards m. You have been given read-only access, so you'll need to create your own copy to follow along. First, create a Miro account for yourself—if you are working in a group, each member must have their own Miro account. Then, one person can copy the Miro board using their account and share it with everyone else. At the time of writing, individual Miro accounts can be created for free, but we cannot predict how Miro as a service or company may evolve. Although making copies and sharing Miro boards is straightforward for tech-savvy users, it may be more challenging if you are unfamiliar with online services. Because of this unpredictability and complexity, we cannot provide support for setting up Miro boards.



1. Getting Started with the Internet of Things

1.1 Introduction and Preparation

Welcome to the first workshop in your journey toward designing and developing Internet of Things (**IoT**) solutions. This is a collaborative exercise intended to deepen your understanding of IoT fundamentals through creative, hands-on exploration. You will work in groups to brainstorm, design, and pitch an IoT concept to one another, gaining practical skills and insights along the way.

Workshop Goals

By the end of this session, you should:

- Know how to select **sensors**, **feedback mechanisms**, and **services** for a user-centric IoT application.
- Understand how to integrate **user needs** and **practical constraints** into an IoT design.
- Communicate your ideas effectively through **storyboards**, **mission statements**, and a **group pitch**.

Pre-Workshop Checklist

To make the most of this workshop, please have:

- The **Tiles IoT Inventor Toolkit** (physical or digital). We recommend visiting the official website (`tilestoolkit.io`) to understand each deck's purpose and usage more thoroughly.
- A **collaborative workspace** such as a whiteboard, sticky notes, or an online board (e.g., Miro) for group ideation.
- **Sketching materials** (pens, markers, blank paper) or a digital drawing platform.

1.2 Objectives

- **Explore and Combine** everyday objects, digital services, and user interface metaphors to conceptualize an *IoT* application.
- **Brainstorm Multiple** (and possibly unconventional) IoT application ideas to broaden your creative thinking.

- **Understand the Core Building Blocks** of IoT systems (sensing, actuation, data handling, user interaction, and services).
- **Design an IoT Solution** that addresses a realistic challenge, emphasizing user needs and group collaboration.

1.3 Learning Outcomes

By completing this workshop, you should be able to:

- **Identify a Problem Context** by applying **Persona** and **Scenario** insights to reflect real user needs.
- **Propose a Coherent IoT System** integrating **Things**, **Sensors**, **Feedback**, **Human Actions**, and **Services**.
- **Create Storyboards** that clarify how the solution functions for both the user and the system.
- **Refine and Reflect** on your design using **Mission** and **Criteria** cards, identifying enhancements and future improvements.
- **Collaboratively Pitch Your Idea** to peers, using a brief, benefits-driven presentation format.

1.4 Related Lessons

Although not mandatory, familiarity with the following lessons (📺) will enrich your understanding:

- **Lesson 1 – Applications and Use Cases**: Real-world IoT deployments (e.g., smart logistics, agriculture) illustrating the breadth of possibilities.
- **Lesson 3 – Sensing and Actuation**: Core principles behind data collection and physical interactions with the environment.
- **Lesson 5 – Data Management and Analytics**: Methods for effectively capturing, storing, and analyzing IoT data.

1.5 Workshop Material Access

Resources You can complete this workshop using:

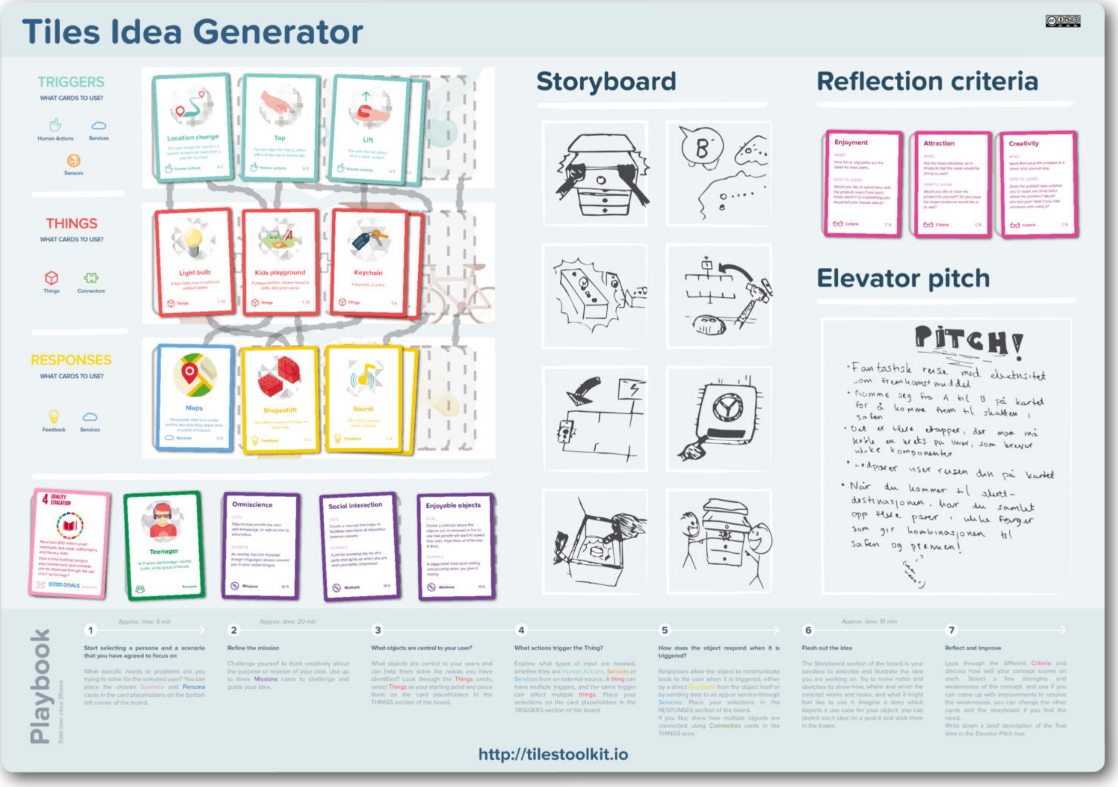
- **Digital Materials**: Access the Miro Board from M.
- **Physical Materials**: Download and print the card sets from ◆.

Credits to the original creators: Tiles Toolkit

1.6 Recommended Outcome of the Workshop

By the end of this workshop, each group should ideally produce:

- A **Storyboard** outlining the user's journey and the system's technical flow.
- A **Problem Context** that is grounded in a **Scenario** card, highlighting user challenges.
- A focused **Mission Statement** describing the key goal of your IoT design.
- A short **Group Pitch** delivered to your peers, explaining the value and feasibility of the concept.



PITCH!

- Fantastisk reise mod elektrisitet som fremkomst middel
- Komme seg fra A til B på kartet for å komme frem til skatten i safen
- Det er flere etapper, der man må koble en krets på hver, som krever ulike komponenter
- ~~·~~dparer user reisen din på kartet
- Når du kommer til slutt-destinasjonen, har du samlet opp flere parer i ulike farger som gir kombinasjonen til safen og premien!

1 Approx. time: 5 min
Start selecting a persona and a scenario that you have agreed to focus on
What specific needs or problems are you trying to solve for the selected user? You can place the chosen Scenario and Persona cards in the card placeholders on the bottom left corner of the board.

2 Approx. time: 20 min
Refine the mission
Challenge yourself to think creatively about the purpose or mission of your idea. Use up to three Missions cards to challenge and guide your idea.

3
What objects are central to your user?
What objects are central to your users and can help them solve the needs you have identified? Look through the Things cards, select Things as your starting point and place them on the card placeholders in the THINGS section of the board.

4
What actions trigger the Thing?
Explore what types of input are needed, whether they are Human Actions, Sensors or Services from an external source. A thing can have multiple triggers, and the same trigger can affect multiple things. Place your selections on the card placeholders in the TRIGGERS section of the board.

5
How does the object respond when it is triggered?
Responses allow the object to communicate back to the user when it is triggered, either by a direct Feedback from the object itself or by sending data to an app or service through Services. Place your selections in the RESPONSES section of the board.
If you like, show how multiple objects are connected using Connectors cards in the THINGS area.

6 Approx. time: 10 min
Flesh out the idea
The Storyboard section of the board is your sandbox to describe and illustrate the idea you are working on. Try to make notes and sketches to show how, where and when the concept works and looks, and what it might feel like to use it. Imagine a story which depicts a use case for your object, you can sketch each step on a post-it and stick them in the boxes.

7
Reflect and improve
Look through the different Criteria and discuss how well your concept scores on each. Select a few strenghts and weaknesses of the concept, and see if you can come up with improvements to resolve the weaknesses, you can change the other cards and the storyboard if you feel the need.
Write down a brief description of the final idea in the Elevator Pitch box.

http://tilestoolkit.io

Figure 1.1: This figure () shows a concise workshop outcome where the group has pinned together their *Persona*, *Scenario*, and *IoT Components* alongside a simple storyboard. Notice how each element (the user's perspective, the system flow, and potential feedback loops) is visually articulated, enabling the team to discuss and refine the concept easily.

Measuring Success

To assess how well your group is meeting the goals of Workshop 1, check if you can answer **yes** to these points:

- Does your storyboard demonstrate *at least two* user interactions and one form of **feedback** (e.g., an alert or display)?
- Have you identified a clear **problem context** connected to a *Persona* and *Scenario* card?
- Can your **mission statement** be summarized in a single sentence, explaining the main value to the user?

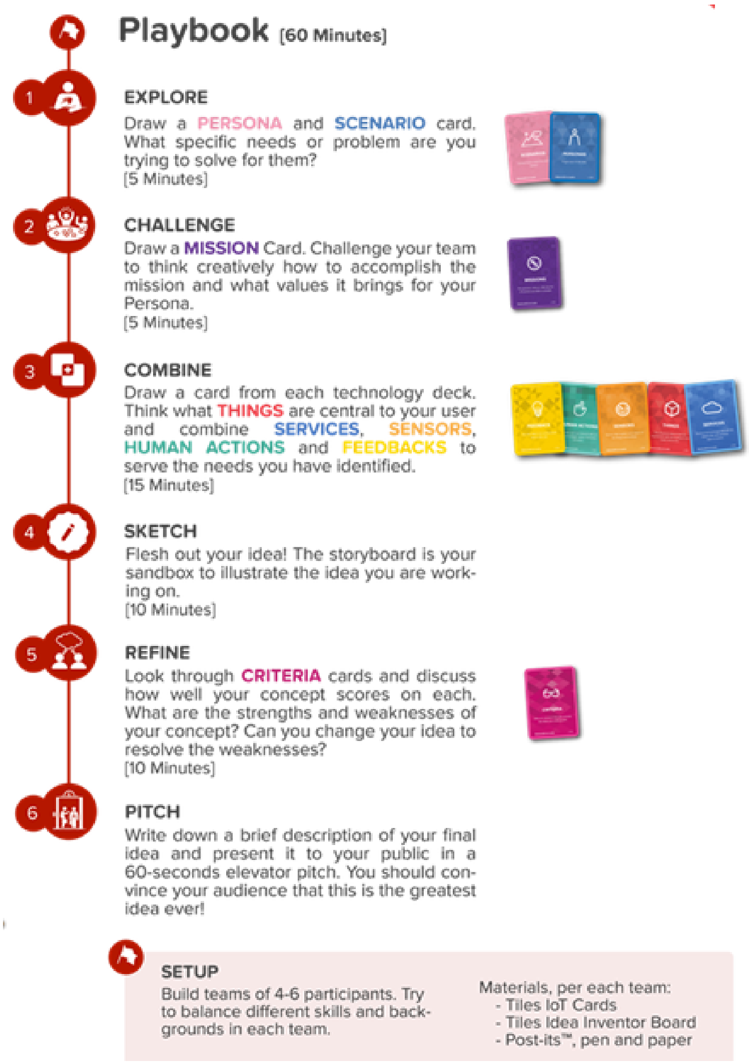
1.7 Quick-Reference Table

The Table 1.1 outlines each main step, a recommended duration, and its output. Refer to this table whenever you need a quick reminder of the activity flow.

Table 1.1: Workshop Steps, Activities, Time, and Outputs

Step	Activity	Time	Main Output
1	Explore Personas & Scenarios	**5–10 min**	Refined problem context
2	Define the Mission	**5–7 min**	Concise mission statement
3	Combine IoT Components	**12–18 min**	System outline
4	Sketch Your Concept	**8–12 min**	Storyboard of user flow
5	Refine w/ Criteria Cards	**15–20 min**	Enhanced concept
6	Pitch Your Solution	**5–7 min**	3-minute pitch (peer feedback)
7	Document & Wrap-Up	**5–8 min**	Photos, notes, reflections

1.8 Process Overview



Playbook [60 Minutes]

1 EXPLORE
Draw a PERSONA and SCENARIO card. What specific needs or problem are you trying to solve for them?
[5 Minutes]

2 CHALLENGE
Draw a MISSION Card. Challenge your team to think creatively how to accomplish the mission and what values it brings for your Persona.
[5 Minutes]

3 COMBINE
Draw a card from each technology deck. Think what THINGS are central to your user and combine SERVICES, SENSORS, HUMAN ACTIONS and FEEDBACKS to serve the needs you have identified.
[15 Minutes]

4 SKETCH
Flesh out your idea! The storyboard is your sandbox to illustrate the idea you are working on.
[10 Minutes]

5 REFINE
Look through CRITERIA cards and discuss how well your concept scores on each. What are the strengths and weaknesses of your concept? Can you change your idea to resolve the weaknesses?
[10 Minutes]

6 PITCH
Write down a brief description of your final idea and present it to your public in a 60-seconds elevator pitch. You should convince your audience that this is the greatest idea ever!

SETUP
Build teams of 4-6 participants. Try to balance different skills and backgrounds in each team.

Materials, per each team:
- Tiles IoT Cards
- Tiles Idea Inventor Board
- Post-its™, pen and paper

Figure 1.2: In this workflow (), you see how each design choice (e.g., selecting a Persona or picking Sensors) impacts subsequent steps. By following the arrows from left to right, you can track the logical progression from problem definition to final concept presentation, ensuring your IoT solution remains consistent and coherent.

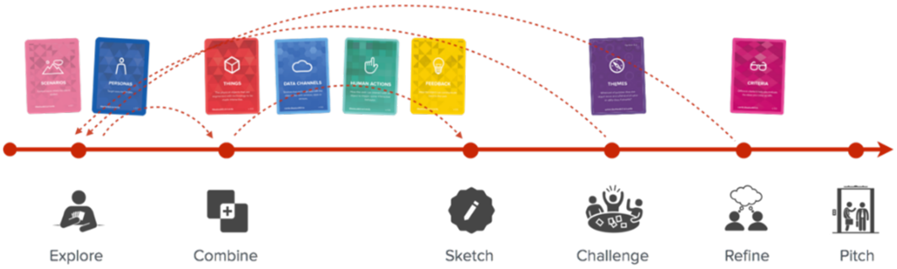


Figure 1.3: This diagram () illustrates a high-level ideation process beginning with clarifying user needs, moving through iterative idea development, and culminating in a basic prototype or concept pitch. Each loop back encourages rethinking and refining the solution before finalizing it.

1.9 Key Building Blocks of IoT Applications

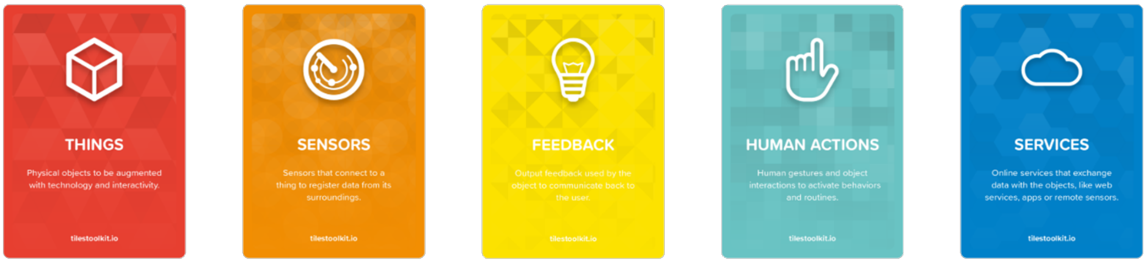

THINGS
Physical objects to be augmented with technology and interactivity.
tilestoolkit.io
SENSORS
Sensors that connect to a thing to register data from its surroundings.
tilestoolkit.io
FEEDBACK
Output feedback used by the object to communicate back to the user.
tilestoolkit.io
HUMAN ACTIONS
Human gestures and object interactions to activate behaviors and routines.
tilestoolkit.io
SERVICES
Online services that exchange data with the objects, like web services, apps or remote sensors.
tilestoolkit.io

Figure 1.4: In addition to the other decks, these **core *Tiles*** () form the building blocks for IoT solutions. *Things (Red)* represent the connected objects; *Sensors (Orange)* gather environmental or contextual data; *Feedback (Yellow)* relays information to users; *Human Actions (Teal)* define how people interact; and *Services (Blue)* handle data storage, analytics, or additional functionality.

Use these elements to shape your overall IoT design. You will also work with **Persona**, **Scenario**, **Mission**, and **Criteria** cards to set the context and refine your ideas. These cards are part of the Tiles IoT Inventor Toolkit described above in section 5.5 and illustrated in Figure 1.5.

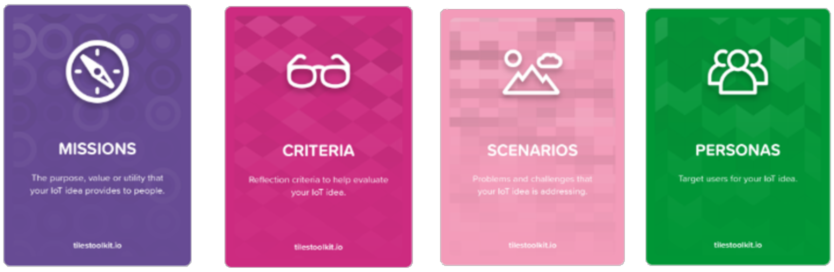

MISSIONS
The purpose, value or utility that your IoT idea provides to people.
tilestoolkit.io
CRITERIA
Reflection criteria to help evaluate your IoT idea.
tilestoolkit.io
SCENARIOS
Problems and challenges that your IoT idea is addressing.
tilestoolkit.io
PERSONAS
Target users for your IoT idea.
tilestoolkit.io

Figure 1.5: In addition to the *Things, Sensors, Feedback, Human Actions,* and *Services* cards, these extra decks () bring depth to your brainstorming. *Mission (Purple)* focuses on overarching goals like sustainability or safety; *Criteria (Pink)* help you evaluate design quality; *Scenarios (Light Pink)* provide real-world contexts such as urban commuting or waste management; and *Personas (Green)* represent target users with unique backgrounds, challenges, or preferences.

1.10 Playbook: Step-by-Step Guide

Below is a walkthrough of the workshop steps. Adjust timings according to your group's pace and interest. Remember: the goal is **exploration and discussion**.

Step 1: Explore the Personas and Scenarios (5–10 Minutes)

Why this Step? Understanding the user's perspective and real-world context from the outset ensures your solution remains *human-centered* and focuses on genuine problems.

- **Draw a Persona Card.** Read the card carefully and discuss the user's needs, daily habits, or constraints.
  For deeper insights, consider spending a few minutes collecting quick user feedback or referencing online communities to validate these needs (e.g., short interviews or surveys).
- **Draw a Scenario Card.** Identify a broader environmental or societal issue (e.g., noise pollution, traffic congestion) that impacts the persona.
- **Combine Insights.** Ask your group: *Which specific problems does our Persona face in this Scenario?* List two or three ways an IoT solution might help.
  For example, if your persona is an elderly user in a high-traffic urban area (Scenario: traffic congestion), you might address air quality or safe navigation by proposing sensors that measure pollution and provide alerts.

> Hints and Tips: If a particular persona-scenario pair does not spark interest, feel free to select another set or invent your own. The objective is to find a context that resonates with your group. **User research can guide you here**: a quick chat with real people in similar situations can confirm whether the chosen scenario matches actual pain points. ■

Step 2: Define the Mission (5–7 Minutes)

Why this Step? Crafting a clear, concise mission statement helps the team focus on the *core goal* of the IoT application, ensuring everyone aligns on the same overarching purpose before selecting components or features.

- **Draw a Mission Card.** Let this statement guide your overall aim (e.g., "enhance personal safety," "promote healthy habits").
- **Value Proposition.** Discuss how the mission ties into both the persona's needs and the scenario's challenges. Write down at least three specific contributions your mission could make.
- **Formulate a Mission Statement.** For instance, "*Help busy families reduce energy consumption by monitoring and automating home appliances in real time*."

> Hints and Tips: Keep your statement concise, making it easy to revisit and ensuring everyone recalls the main objective. ■

Step 3: Combine IoT Components (12–18 Minutes)

Why this Step? Choosing appropriate *Things*, *Sensors*, *Feedback*, *Human Actions*, and *Services* early on sets the technical foundation. It ensures your idea is feasible and addresses the user needs identified in previous steps.

- **Things.** Select a red *Things* card (e.g., a *streetlamp*, *refrigerator*, or *bicycle*) or invent a new object that could be "smartified."
- **Sensors.** Pick an orange *Sensors* card aligning with your mission. Examples: temperature, light, CO_2, accelerometer.
- **Feedback.** Use a yellow *Feedback* card to determine how the system communicates (LED display, smartphone alert, audible alarm).

- **Human Actions.** Incorporate a teal *Human Actions* card specifying how users interact (voice commands, gesture control, mobile app).
- **Services.** Finish by choosing a blue *Services* card for data analysis, cloud storage, or social integration.
- **Sketch the Data Flow:** Draw a rough block diagram indicating how data travels from sensors to services and how feedback returns to the user.

Hints and Tips:

- **Unconventional Sensors.** Think about sensors for noise, vibration, soil moisture, or even heart rate to stand out from typical choices.
- **Accessibility.** If your Persona has specific requirements (e.g., visual impairment), consider tactile or auditory feedback over visual indicators.
- **Multifunctional Items.** A single device can do more than one job—like a *smart bin* that measures waste levels and guides recycling behaviors.

■

Step 4: Sketch Your Concept (8–12 Minutes)

Why this Step? Visual storyboards clarify how users will interact with your IoT system in real-life scenarios. This helps spot potential gaps or complications in the design before finalizing any technical details.

- **Storyboard the User Flow.** Create a series of 3–5 panels showing how the user interacts with the IoT solution, from setup to receiving feedback.
- **Consider Environment.** If it's outdoors, factor in weather or noise; if indoors, think about Wi-Fi and layout constraints.
- **Label Components.** Indicate where sensors collect data, how services process it, and how the user sees or hears the feedback.

Hints and Tips: Simple sketches are sufficient. The priority is clarity of the concept rather than artistic detail. ■

Step 5: Refine Your Concept Using Criteria Cards (15–20 Minutes)

Why this Step? Considering criteria like *usability*, *cost*, *privacy*, or *sustainability* helps improve the design. By looking at weak spots early, you can avoid bigger issues down the line.

- **Draw a Criteria Card.** Examine your design through lenses such as *usability*, *cost*, *sustainability*, or *privacy*.
- **Identify Weak Points.** Could it become too expensive? Is security or privacy adequately addressed? Are you collecting more data than necessary?
- **Iterate.** Make small adjustments—like swapping a high-cost sensor for a budget-friendly one or ensuring data is encrypted to build trust.

Hints and Tips: Talk about how your system might operate under load or across multiple users. This often reveals scalability concerns early. ■

Step 6: Pitch Your Solution (5–7 Minutes)

Why this Step? A concise group pitch helps you *sell* your concept and gather feedback quickly. It ensures everyone understands the essential user problem, core IoT setup, and main benefits.

- **Group Pitch.** Present your solution in about three minutes to other groups. Focus on clarifying the user problem, how your IoT system works, and the main benefits.
- **Peer Feedback.** Ask the audience for one praise point (e.g., "I love the creative sensor choice!") and one constructive critique (e.g., "How would it work in rural areas with no Wi-Fi?").

Step 7: Document and Wrap-Up (5–8 Minutes)

Why this Step? Capturing final outputs and team reflections consolidates learning. It also provides a record you can revisit or expand on in future workshops.

- **Capture Everything.** Take photos or screenshots of your storyboard, cards, and any notes.
- **Reflect as a Team.** Discuss key learnings, surprising discoveries, and potential improvements for a future iteration.
- **Group Portfolio.** Organize these materials (images, sketches, bullet-point notes) into a shared folder or presentation, so you can revisit your ideas later.

> Hints and Tips: Reflection often unveils ideas you might have missed in the rush of brainstorming. Spend a moment letting each member share their takeaway. ■

1.11 Common Pitfalls

- **Overloading the Design.** Resist the urge to add every feature at once; start lean and essential.
- **Forgetting the User.** Always circle back to the persona's perspective to ensure the solution remains people-focused.
- **Ignoring Security/Privacy.** Even conceptually, addressing data protection fosters user trust.
- **Unclear Pitch.** Make sure your short presentation clearly states the *problem*, the *IoT setup*, and its *benefits*.
- **Collaboration Imbalances.** Beware of one or two voices dominating the conversation. Ensure each member (especially quieter participants) can contribute ideas and receive constructive feedback. A designated *facilitator* or *moderator* can help keep discussions balanced.
- **Skipping Group Feedback.** Under time pressure, it's tempting to finalize an idea without hearing from the entire team. This risks missing valuable insights or uncovering potential flaws in your concept.

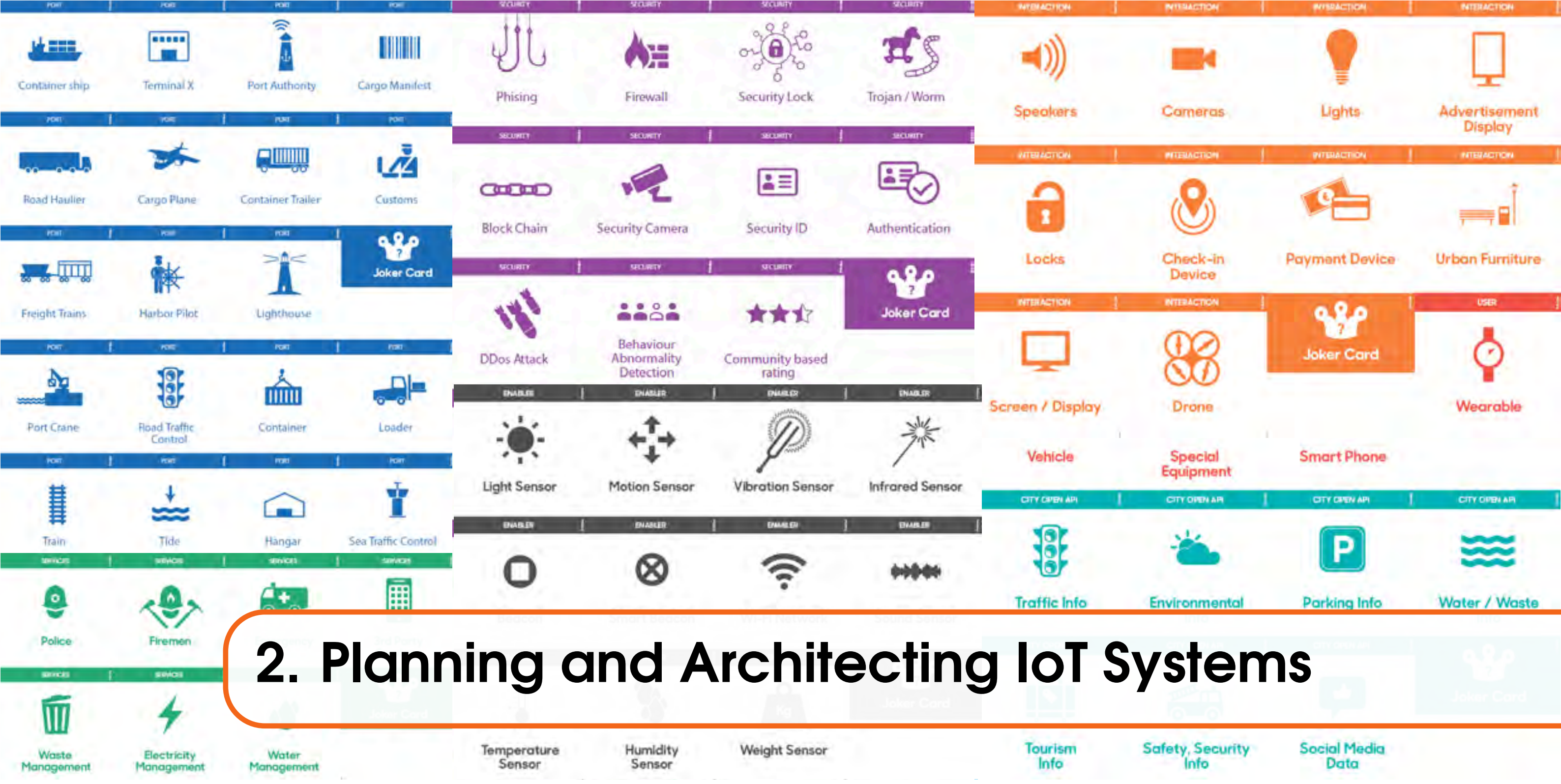
1.12 Conclusion

Workshop 1 served as your introduction to **user-centric IoT design**, guiding you through the fundamentals of brainstorming and concept development. By combining *Personas, Scenarios, Missions*, and core **IoT building blocks**, you explored how everyday objects can become "smart" solutions that address real user needs. Through collaboration and creative thinking, you established a strong foundation that will support more advanced technical considerations in later workshops. As you move forward, remember that this experience in ideation, feedback loops, and user-driven problem solving will be invaluable—whether you choose to pursue further workshops or simply carry these insights into your future projects.

Looking Ahead. As you move on to **Workshop 2**, you will take the concepts and storyboards created here and refine them by exploring *IoT architectures, networking, and communication flows*. This ensures a seamless progression from idea generation to a more robust system design.

Reflection and Next Steps. We encourage you to schedule a brief "show & tell" or "demo session" after finalizing your initial concept. Use this opportunity to share key takeaways, lessons learned, and any potential changes you plan to implement. This reflection helps solidify your progress and lays the foundation for future workshops or deeper research.

Real-World Inspiration. Consider existing *IoT solutions* for reference—like smart bins used in various European cities to optimize waste collection, or connected streetlights that automatically dim when no motion is detected. Seeing how similar problems are addressed in practice can spark new ideas or highlight practical constraints (e.g., power consumption, network coverage) you may need to address.



2. Planning and Architecting IoT Systems

2.1 Introduction and Preparation

Welcome to **Workshop 2: Planning and Architecting IoT Systems**. This workshop builds upon the conceptual ideas you explored in *Workshop 1*, with a deeper focus on **system architecture**, **networking**, and **communication** for robust IoT solutions.

You can approach this workshop in two ways:

1. *Develop Further*: If you completed Workshop 1, bring *your existing storyboards, scenarios, or user needs* and refine those ideas by focusing on the architecture layer.
2. *Fresh Start*: Alternatively, if you are joining directly, you can create a new IoT concept in this session, using the same system-level steps described here.

Either path will benefit from a user-centric focus and the practical techniques outlined below. Like the previous workshop, there is **no formal evaluation**; instead, you will collaborate to expand your understanding of IoT systems and share insights through group activities.

Workshop Goals

By the end of this session, you should be able to:

- **Explore Concepts**: Investigate how physical objects, digital services, and user experiences intersect in an IoT architecture.
- **Generate Ideas**: Brainstorm multiple system-level possibilities, focusing on scalability, reliability, and connectivity.
- **Understand Core Components**: Recognize the main building blocks (maps, tokens, cards) needed to visualize and plan IoT systems.
- **Apply Design Thinking *Iteratively***: Use the *IoT Service Kit* to design a practical IoT solution, emphasizing user needs, interactions, and security considerations.

Pre-Workshop Checklist

Before starting, ensure you have:

- **IoT Service Kit** (physical or digital) including maps, tokens, and cards (Sensors, Interactions, Services, Open APIs, User, etc.).

- **Collaboration Space** such as a whiteboard, sticky notes, or an online board for group brainstorming and system mapping.
- **Basic Understanding** of IoT fundamentals from Workshop 1 (or a similar baseline), especially around conceptual design, user research, and scenario-building.

2.2 Objectives

- **Evaluate and Refine** existing IoT concepts by focusing on system-level design.
- **Consider Architectural Patterns** to choose frameworks suited to your application's scale and complexity.
- **Analyze Networking Protocols** and communication flows, aligning them with environment constraints.
- **Develop a Coherent Architecture** that integrates sensing, data flow, interactions, and services securely and efficiently.

2.3 Learning Outcomes

By completing this workshop, you should be able to:

- **Propose a System Architecture** for your IoT solution, showing how components connect and communicate.
- **Identify Key Networking Decisions** relevant to your chosen environment (e.g., Wi-Fi vs. LoRaWAN, edge vs. cloud).
- **Assess Architectural Trade-offs** (latency, cost, security) when selecting hardware, software, and networking protocols.
- **Outline a Data Flow** that maps how information moves between sensors, services, and users, ensuring reliability and scalability.
- **Integrate Security Considerations** (encryption, authentication) into your architectural plan.
- **Verify Architecture Against User Constraints**: Confirm it aligns with real-world budgets, environment challenges, and operational needs.

2.4 Related Lessons

Although not mandatory, familiarity with the following lessons (📺) will enrich your understanding:

- **Lesson 2 – Architectures**: Covers basic IoT architectures (client-server, publish-subscribe, edge-cloud) and typical design patterns.
- **Lesson 4 – Networking and Communications**: Delves into network protocols, communication models, and constraints affecting IoT devices.

2.5 Workshop Material Access

Resources **Choose your resources:**

- **Digital Materials**: Access the Miro Board from m.
  - *Note*: Free plans on Miro may limit certain features (e.g., board sizes), so plan accordingly or consider an alternative board if needed.
- **Physical Materials**: Download and print the card sets from ◆.

Credits to the original creators: IoT Service Kit

2.6 Recommended Outcome of the Workshop

By the end of this workshop, your group should have:

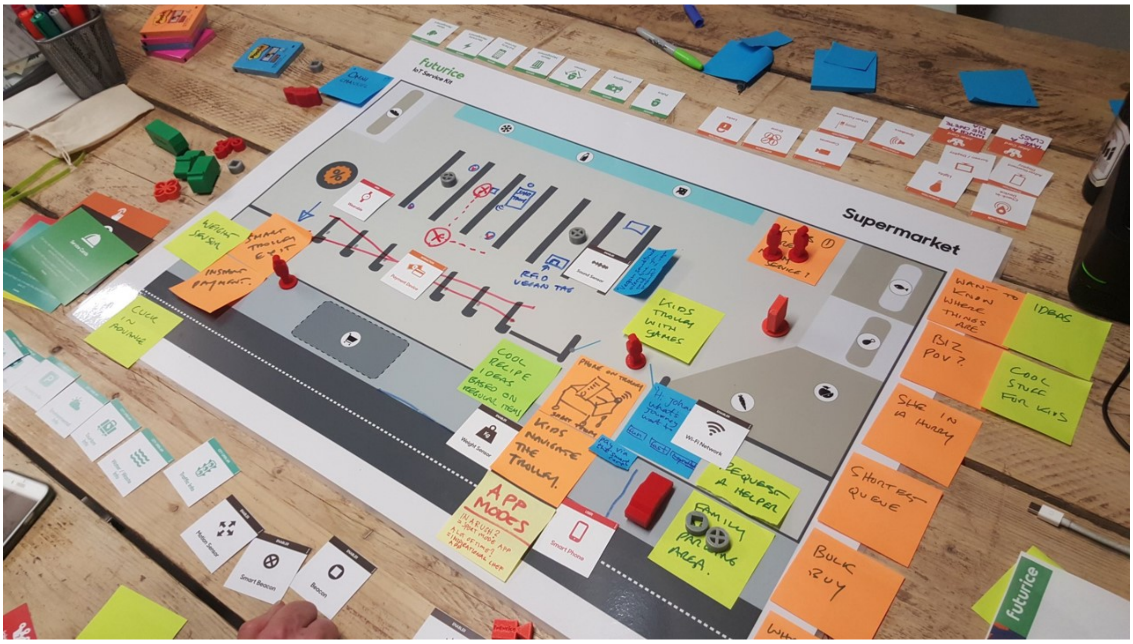


Figure 2.1: Workshop Output featuring an interactive layout, markers for user journeys, and scattered brainstorming notes. Each physical artifact can be quickly repositioned or replaced, enabling continuous, collaborative shaping of your IoT ecosystem. This flexible setup mirrors the iterative nature of user-centric design. Additionally, such real-world mapping encourages reflection on possible constraints (e.g., Wi-Fi dead zones, limited budgets) or user needs discovered in previous workshops. **Consider sharing your final outcome** with other groups or mentors for early feedback.

- **A Detailed IoT Architecture**: Illustrated using maps, tokens, and relevant cards.
- **Key Networking Decisions Documented**: Including protocol choices, connectivity strategies, and trade-off discussions.
- **Data Flow Diagram**: Showcasing how sensor data travels, is processed, and feeds back to users or other services.
- **Security and Privacy Plan**: Identifying potential vulnerabilities and basic mitigation strategies.

Measuring Success

Success Measures: As you reach the end of this workshop, consider:

- Have you clearly documented *which network protocol(s)* are used and *why*?
- Does your architecture diagram illustrate **data flow** from sensors to services, including any **edge** or **cloud** components?
- Have you listed at least **one security safeguard** (e.g., encryption, authentication) relevant to your environment?
- Did you address **user constraints** or environment-specific limitations (e.g., limited bandwidth, intermittent connectivity)?
- *Have you sought input or feedback* from peers or stakeholders to catch gaps early?

■

2.7 Quick-Reference Table

Table 2.1: Workshop 2 Steps, Activities, Time, and Outputs (detailed)

Step	Activity	Time	Main Output
1	Setting the Scene – Choosing the Environment	5 min	Selected environment map and purpose statement (including user constraints)
2	Focus on the Journey – Tokens and User Paths	10 min	Mapped user flow with tokens, identified friction points
3	Design Interactions and Integrations	15 min	Placement of sensors, services, and user interactions
4	Architecture Deep Dive – Networking and Data Flow	15 min	Network protocols chosen and data flow diagram (reference section 2.8 for patterns)
5	Security and Privacy Considerations	10 min	Identified vulnerabilities, basic mitigation strategies
6	Consolidate and Document Your Architecture	5 min	Final architecture outline and short group presentation (testing steps encouraged)

Note on Timing Flexibility: If you have limited time, consider merging Steps 3 and 4. If you want deeper security coverage, allocate more time to Step 5.

2.8 Additional Guidance: IoT Architecture Patterns

Note: Before completing Step 4 (Architecture Deep Dive), you may wish to review the following typical *IoT architecture patterns* to see which best fits your needs. Many IoT solutions can be categorized by patterns like:

- **Client-Server Model**: Sensors or devices act as clients, making direct requests to a central server.
  - *Advantages*: Simple to implement for fewer devices and low real-time needs.
  - *Drawbacks*: Can become a bottleneck or single point of failure at scale.
  - *Example*: A small lab with a handful of sensors periodically sending data to a single gateway PC.
- **Publish-Subscribe (e.g., MQTT)**: Devices publish messages to a broker, and subscribers receive only topics they need.
  - *Advantages*: Decouples senders from receivers, scalable, ideal for event-driven systems.
  - *Drawbacks*: Requires a broker; more complex than simple request-response.
  - *Example*: A greenhouse where multiple sensors publish environmental data, while separate apps subscribe for real-time alerts.
- **Edge-Cloud or Hybrid**: Some computing occurs at the edge (e.g., gateways or microcontrollers) before sending data to the cloud for heavier analytics.
  - *Advantages*: Reduces latency, saves bandwidth, improves local responsiveness.
  - *Drawbacks*: Edge devices can be more expensive or complex to manage.
  - *Example*: A manufacturing plant with local edge computing for quick safety checks, uploading summarized data to the cloud.
- **Hub-and-Spoke or Gateway-Based**: A local gateway manages communication with devices and the external cloud.
  - *Advantages*: Simplifies device management; the gateway can handle multiple protocols (Zigbee, Bluetooth, etc.).
  - *Drawbacks*: The gateway is a single point of failure if not made redundant.
  - *Example*: A smart home with one “hub” bridging sensors and a cloud platform.

Use these patterns to guide your decision-making when **choosing network protocols** or **implementing edge solutions**. The best pattern often emerges from balancing functional needs (real-time monitoring vs. periodic updates) and practical constraints (cost, power, coverage area). Consider how each choice may tie in with **non-functional requirements**—such as *latency*, *security*, or *scalability*—which are discussed in later modules.

2.9 Introducing the IoT Service Kit

The **IoT Service Kit** functions like a collaborative board game, supporting multidisciplinary teams in designing **user-centric IoT experiences**. It simplifies system thinking with tangible maps, tokens, and cards, making advanced concepts easier to handle even for participants without deep technical backgrounds.

How the IoT Service Kit Helps

- **Achieve Mutual Understanding**: Break down communication barriers between different expertise areas.
- **Stay Tangible**: Transform abstract IoT systems into physical layouts, helping teams visualize and iterate quickly.
- **Simplify the Complex**: Provide an approachable entry point for analyzing architecture, networking, and security, without overwhelming non-technical stakeholders.

Components of the IoT Service Kit

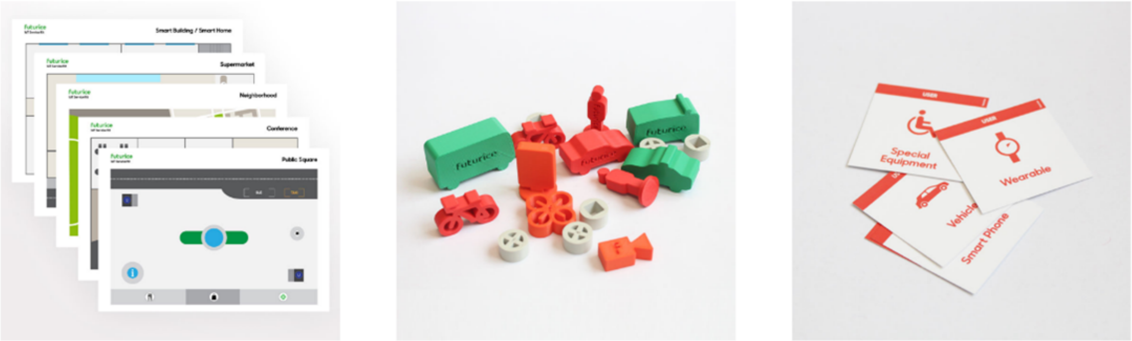


Figure 2.2: These images display various references for IoT ideation. The left frame showcases environment-specific "boards," like smart buildings or supermarkets. The central frame highlights 3D tokens and connectors for simulating physical objects or infrastructure. On the right, sample user-equipment cards illustrate the hardware requirements (e.g., smartphones, wearables) that individuals might need to interact with your IoT solution.

1. **Maps**: Represent environments like smart homes, offices, supermarkets, or ports. (Figure 2.2 shows an example of various boards and tokens.)
2. **Tokens**: Physical markers (e.g., 3D printed icons) to represent users, devices, vehicles, or infrastructure. (See Figure 2.10 for a glimpse of a real workshop.)
3. **Cards**:
   - **Sensors**: Indicate data collection points (e.g., motion, temperature, camera). See Figure 2.3(a).
   - **Interactions**: Highlight user interaction forms (touchscreens, voice commands). See Figure 2.3(b).
   - **User Cards**: Define user roles, permissions, and required equipment. See Figure 2.3(c).
   - **Service Cards**: Suggest potential integrations (payment systems, environmental monitoring). See Figure 2.3(d).

- **Security**: Safeguard the system with encryption, authentication protocols, and threat modeling. See Figure 2.3(e).
- **Open APIs**: Reference publicly available data sources (e.g., city traffic, weather). See Figure 2.3(f).
- **Ports**: A smart port scenario showcases the complexities of industrial-scale IoT. From logistics and cargo handling to transportation tracking, such an environment demands robust, secure, and often large-scale connectivity solutions. See Figure 2.4.

2.10 Step-by-Step Guide

Below is a structured approach to refining your IoT solution architecture using the **IoT Service Kit**. Adjust timings and level of detail based on your group's familiarity with networking concepts.

Step 1: Setting the Scene – Choosing the Environment (5 Minutes)

Start by identifying the environment in which your IoT system will operate. For instance, Figure 2.5 provides a bare-bones supermarket floor plan, while Figure 2.4 highlights a smart port scenario.

- **Select or Create a Map**: Pick a pre-designed map (smart supermarket, city street, harbor port) or sketch one if your scenario is unique.
- **Define the Purpose**: Outline the main goals of the IoT application—monitoring environmental conditions, optimizing traffic flows, enhancing user experience, etc.
- **Consider Environment Constraints**: Indoor vs. outdoor, open areas vs. constrained spaces, and existing infrastructure (e.g., Wi-Fi availability, power sources).
- **Reflect on Real-World Constraints**: Budget, user demographics, or maintenance. If you have user feedback from Workshop 1, incorporate those findings here.

> **Prompt** How might the environment shape your architectural and networking choices (e.g., low-power wide area vs. local mesh)? Have you consulted any real user constraints or cost limitations? ■

Step 2: Focus on the Journey – Tokens and User Paths (10 Minutes)

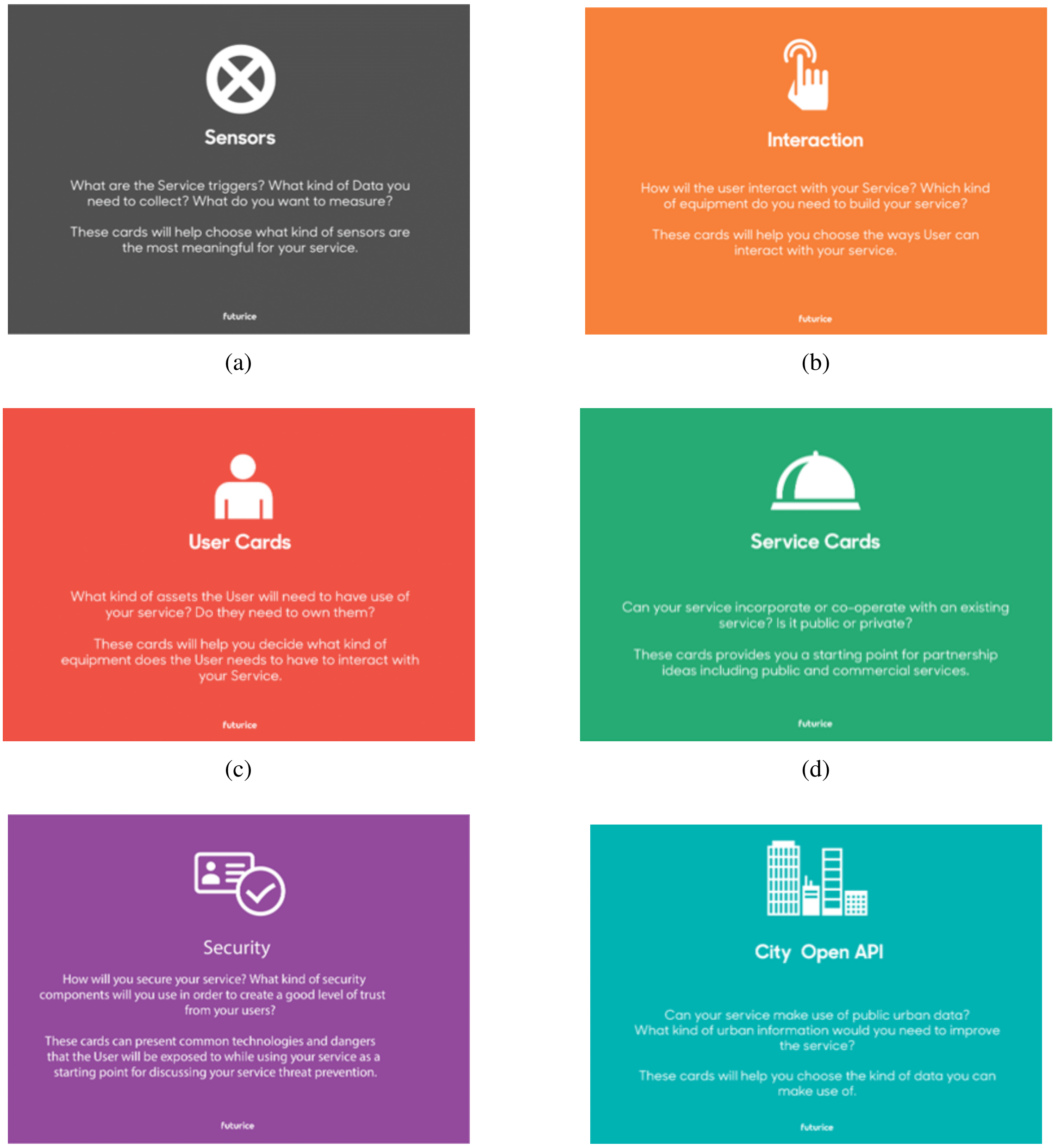
Next, plan out how users and assets move through the chosen space. For example, Figure 2.6 shows a typical customer route in a supermarket, while Figure 2.7 zooms in on a beacon-based interaction. You might also role-play or narrate a day-in-the-life scenario to uncover friction points.

- **Identify Key Participants**: Users (customers, staff, managers) and moving assets (vehicles, deliveries).
- **Place Tokens**: Arrange tokens to depict how users or objects traverse the environment. This helps visualize data-capture points (sensors) and interaction points (displays, check-ins).
- **Friction Points**: Look for areas where communication could fail (dead zones) or where user experience might degrade (long queues, complex navigation).

> **Questions**
>
> - Where do users spend the most time? Could a sensor or interactive display help them here?
> - Are any user paths too complex, suggesting the need for better wayfinding or automation?
>
> ■

Step 3: Design Interactions and Integrations (15 Minutes)

At this stage, you define how sensors, services, and user interactions come together. Figure 2.3(e) underscores the importance of secure components, while Figure 2.3(d) highlights possible service integrations. Keep in mind that each new integration could introduce privacy or security implications.



Sensors

What are the Service triggers? What kind of Data you need to collect? What do you want to measure?

These cards will help choose what kind of sensors are the most meaningful for your service.

futurice

(a)

Interaction

How wil the user interact with your Service? Which kind of equipment do you need to build your service?

These cards will help you choose the ways User can interact with your service.

futurice

(b)

User Cards

What kind of assets the User will need to have use of your service? Do they need to own them?

These cards will help you decide what kind of equipment does the User needs to have to interact with your Service.

futurice

(c)

Service Cards

Can your service incorporate or co-operate with an existing service? Is it public or private?

These cards provides you a starting point for partnership ideas including public and commercial services.

futurice

(d)

Security

How will you secure your service? What kind of security components will you use in order to create a good level of trust from your users?

These cards can present common technologies and dangers that the User will be exposed to while using your service as a starting point for discussing your service threat prevention.

futurice

(e)

City Open API

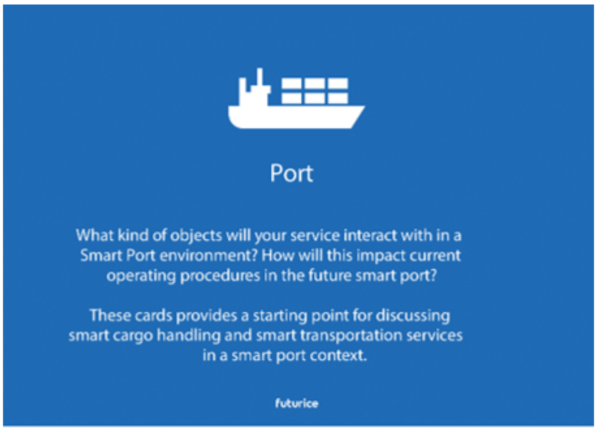
Can your service make use of public urban data? What kind of urban information would you need to improve the service?

These cards will help you choose the kind of data you can make use of.

futurice

(f)

Figure 2.3: (a) An example *Sensors* card from the IoT Service Kit, prompting participants to identify triggers, decide on which data to collect, and choose appropriate sensors. (b) A reminder of how users engage with the IoT solution, prompting consideration of various inputs (touch, voice, RFID) and outputs (notifications, alarms, displays). (c) Emphasizes the user's perspective, clarifying requirements (e.g., smartphone or special badge) to ensure a practical, inclusive design. (d) Highlights potential integrations with public or private services (e.g., city APIs, third-party payments) to enhance functionality for end users. (e) Stresses the importance of security considerations such as encryption standards, authentication protocols, and threat modeling early in the design. (f) Demonstrates how city-level open APIs or other public data sources can enrich an IoT solution with real-time information about traffic, environment, or public services.



Port

What kind of objects will your service interact with in a Smart Port environment? How will this impact current operating procedures in the future smart port?

These cards provides a starting point for discussing smart cargo handling and smart transportation services in a smart port context.

futurice

Figure 2.4: A smart port scenario showcases the complexities of industrial-scale IoT. From logistics and cargo handling to transportation tracking, such an environment demands robust, secure, and often large-scale connectivity solutions.

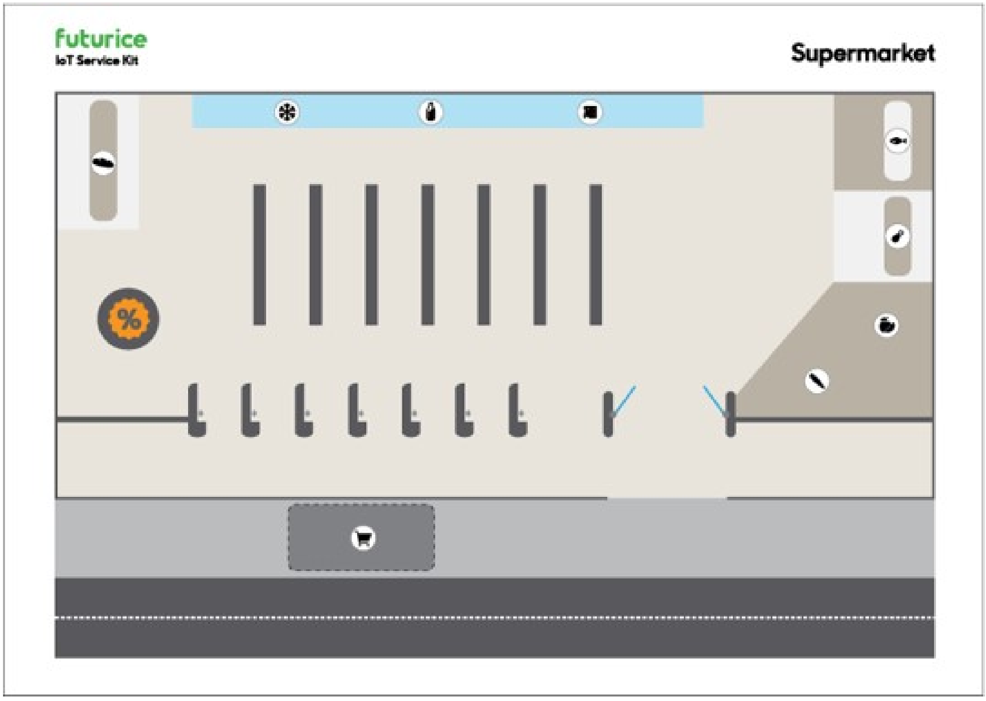


Figure 2.5: Here, a bare-bones supermarket floor plan marks aisles, checkout stations, and product zones. This layout allows you to place sensors and interactive spots where they are most effective, prompting early thought on user flow and data capture in a typical retail setting. Remember to consider *user research* from Workshop 1 or new feedback if you are starting fresh.

- **Use Cards to Represent Interactions**: Mark points where users interact with the system—doors with locks, check-in counters, temperature controls, voice assistants (see Figure 2.3(b)).
- **Incorporate Sensors and Services**: Decide which sensor cards to place (e.g., motion, temperature, camera) and which external or internal services to integrate (e.g., payment APIs, city data).
- **Open API Exploration**: If external data enhances your service, place Open API cards (see Figure 2.3(f)) to visualize how real-time city or weather data might feed into your system.

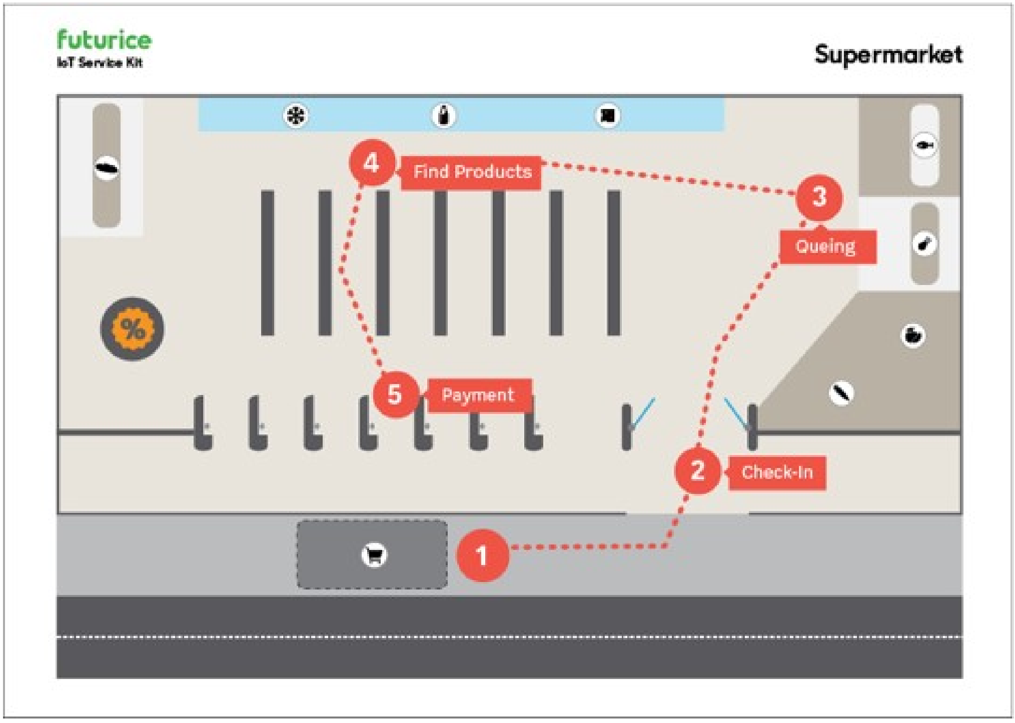


Figure 2.6: Building on the previous diagram, this map overlays a sample customer route: entering, queueing, finding products, and paying. Analyzing step-by-step paths helps you pinpoint possible bottlenecks or confusion, guiding you to design suitable IoT interventions at each stage. In large-scale deployments, also consider *non-functional requirements* like performance under heavy user load.

> **Prompt** How can each interaction improve the overall user experience or add value to the system? Are you factoring in user feedback or real operational constraints? ■

Step 4: Architecture Deep Dive – Networking and Data Flow (15 Minutes)

Once you have mapped interactions, think about the architecture and data flow. For reference, Figure 2.8 and Figure 2.9 show layered IoT solutions in a supermarket context, emphasizing how data moves from capture to analysis. Review section 2.8 now if you haven't yet, then update your initial diagram.

- **Choose Network Protocols**: Reflect on environment constraints to decide among Wi-Fi, Bluetooth, cellular, LoRaWAN, or a hybrid approach.
- **Map Out Data Flow**: Indicate how data travels from sensors to a gateway, then to the cloud or local edge. Show where caching or processing might occur.
- **Latency and Reliability**: Note which components demand real-time (low-latency) connections (e.g., alarms) vs. which can tolerate delay (e.g., batch analytics).

> **Idea Check** Could your system benefit from edge computing or a distributed architecture to reduce bandwidth usage? How do these decisions feed into *non-functional requirements* like performance or fault tolerance? ■

Step 5: Security and Privacy Considerations (10 Minutes)

As illustrated in Figure 2.3(e), safeguarding user data and system operations is integral to IoT design. Be sure to consider user awareness or data usage policies if handling personal information.

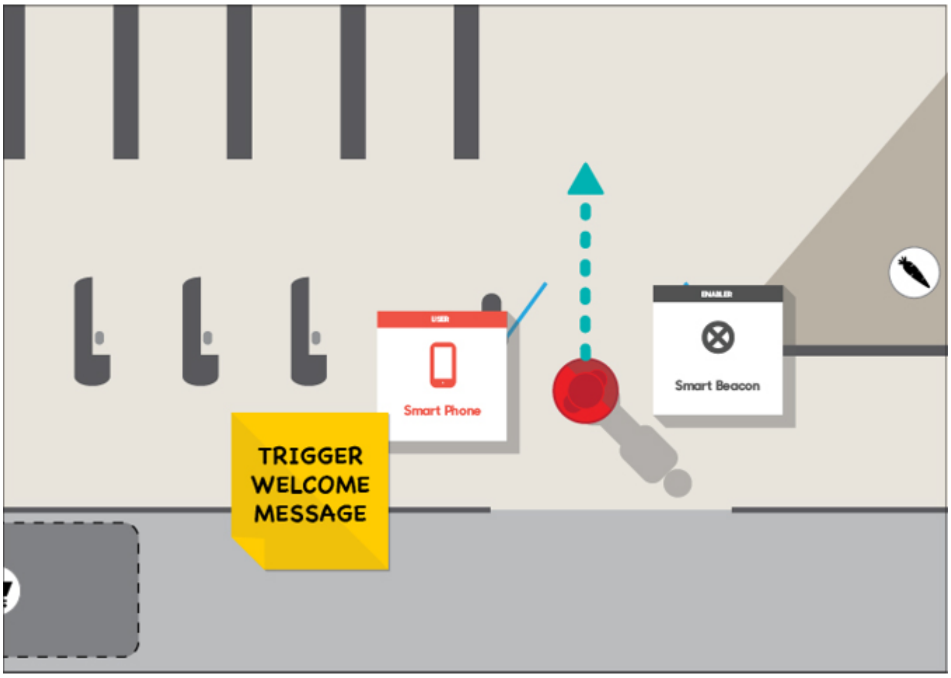


Figure 2.7: A closer look at a single interaction spot shows how a beacon and a user's smartphone might communicate, triggering personalized greetings or product suggestions. Zooming in on such details clarifies data flows (e.g., location tracking, short-range transmissions) and illustrates how to enhance on-site user experiences.

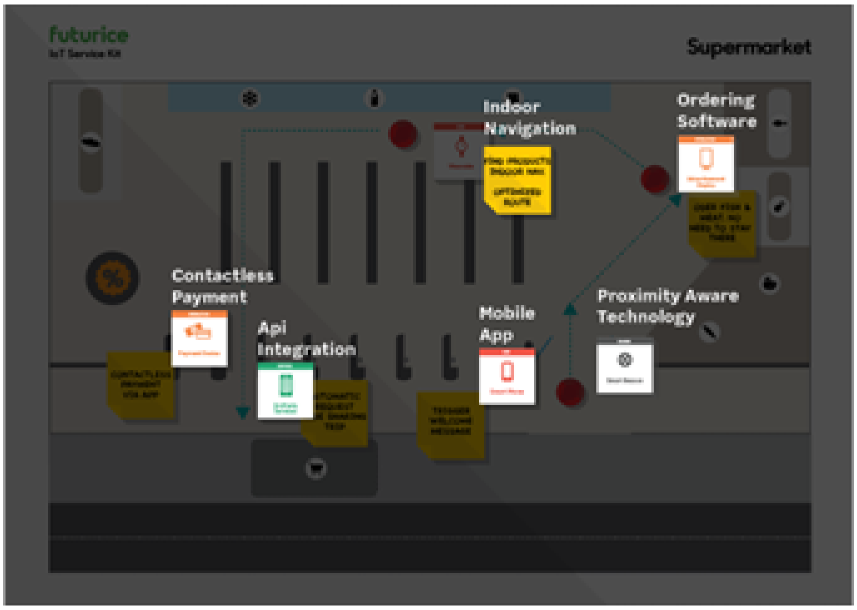


Figure 2.8: This extended supermarket layout layers multiple IoT elements—contactless payment kiosks, indoor navigation, proximity beacons—showing how each connects to form a cohesive ecosystem. Labels like "Ordering Software" or "API Integration" highlight both user-facing and back-end services working in concert. Real-world case studies often cite similar setups, like large retailer analytics or hospital asset tracking.

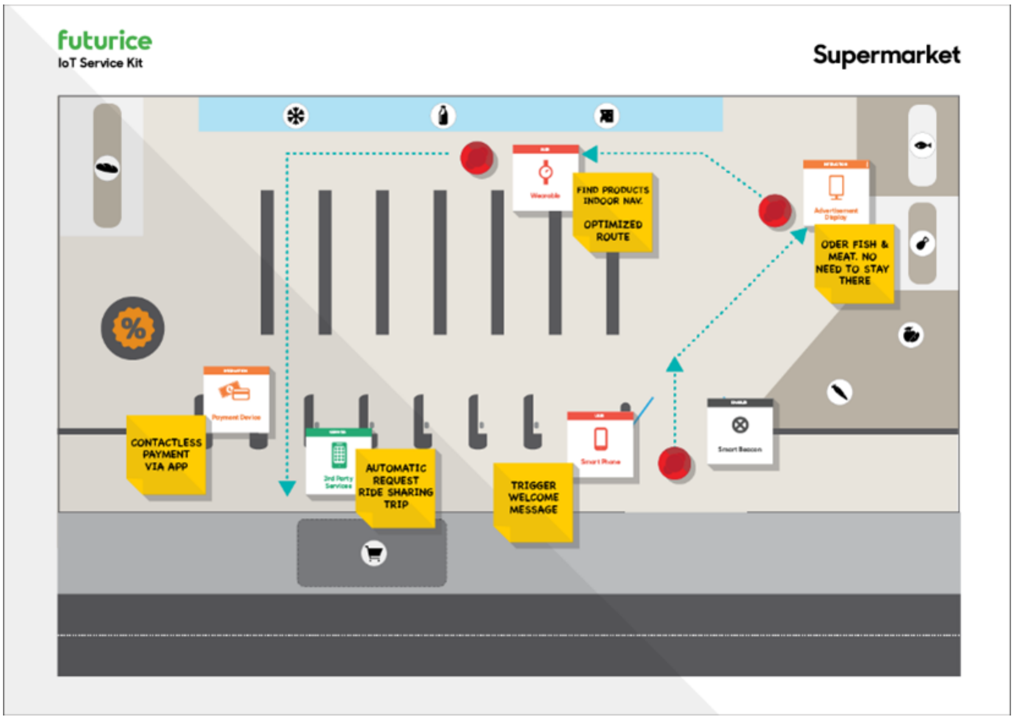


Figure 2.9: In this data-flow perspective, boxes labeled "Customer Location" or "Routes Taken" illustrate actionable insights for store layout optimization, while "People Counter" or "Statistical Data" point to aggregated metrics guiding restocking decisions. Visualizing these inputs and outputs underscores the importance of analytics and feedback loops, which relate directly to *scalability* and *maintenance*—key non-functional requirements.

- **Identify Vulnerabilities**: Data in transit vs. data at rest, user authentication, device credentials.
- **Choose Security Layers**:
  - *Basic*: TLS/SSL for transport encryption, simple password authentication.
  - *Enhanced*: Rotating keys, device certificates, two-factor authentication.
  - *Advanced*: Hardware-based secure elements, on-device intrusion detection.
- **Risk Mitigation**: Brainstorm fail-safes if connectivity fails, or if sensors are tampered with.

> Prompt What minimal security measures are essential to prevent data breaches, especially for sensitive information? ■

Step 6: Consolidate and Document Your Architecture (5 Minutes)

Finally, bring everything together. Figure 2.10 and Figure 2.1 show how participants use sticky notes, tokens, and boards to finalize their designs.

- **Summarize the Architecture**: Create a simplified diagram linking user journeys, sensor placements, network choices, and data flows.
- **Highlight Trade-offs**: Note any compromises (e.g., Wi-Fi for indoor coverage vs. 5G for mobility, cost vs. reliability).
- **Prepare a Short Group Pitch**: Explain how your design addresses user needs, system requirements, and security considerations.

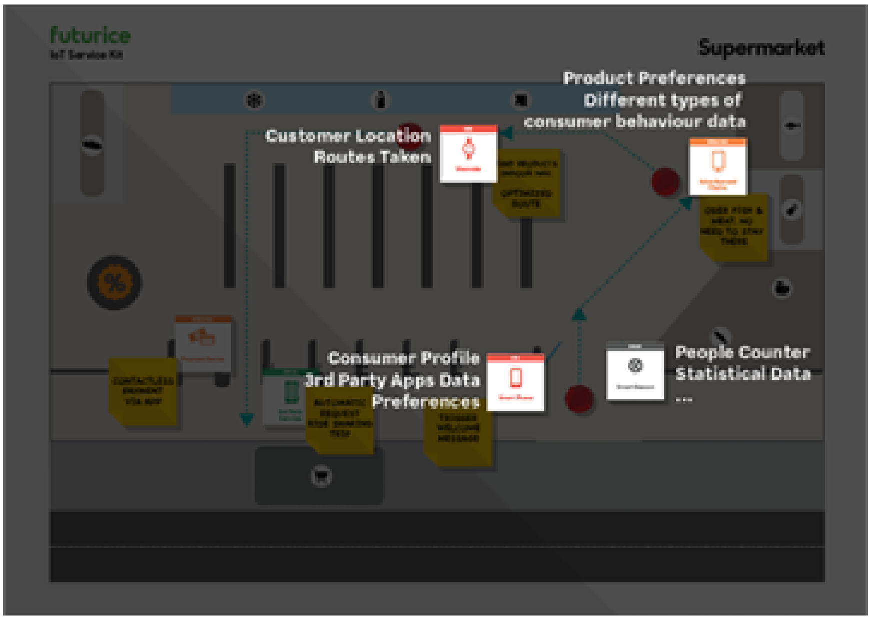


Figure 2.10: A snapshot from a real workshop scenario. Sticky notes, tokens, and user cards bring the supermarket board to life, capturing quick ideas like “Shorter Queues” or “Bulk Buying.” This hands-on approach empowers participants to explore solutions fluidly, iterating as they refine interactions and features. Consider small-scale tests or simulations at this stage to validate assumptions.

2.11 Additional Ideas and Scenarios

Once your core design is set, consider:

- **Integrating with Existing Systems**: Payment gateways, city open data, inventory management, or sensor networks.
- **Reliability and Redundancy**: Edge devices for local data processing when connectivity is unstable.
- **Privacy and Ethics**: Ensure the data you collect is necessary and stored responsibly. Are users informed about data usage?
- **User Feedback and Reporting**: Mechanisms for users to report system issues or anomalies, enabling continuous improvement.
- **Sustainability**: Incorporate energy-saving strategies or consider hardware that minimizes waste and power consumption.
- **Real-World Case Inspiration**: Look at known IoT deployments (e.g., large warehouses using automated guidance vehicles, or stadiums with crowd monitoring). Reflect on how they solve reliability or scale challenges.
- **Business Angle**: If relevant, consider a brief value proposition or cost model—could this scale to a commercial deployment?

2.12 Activity Reflection

Reflecting on your design process can help solidify lessons learned and highlight areas for improvement:

1. **Review User Experience**: Does your architecture genuinely serve user needs, or does it impose friction (e.g., complicated setup, extra hardware)?
2. **Reevaluate Tech and Data Choices**: Are there simpler, cheaper, or more secure alternatives to the chosen sensors and protocols?

3. **Security and Privacy Check**: Have you minimized data collection and integrated robust authentication or encryption?
4. **Are You Testing Iteratively?**: Consider running a small simulation or pilot to check if your chosen protocols handle real-world traffic volumes.
5. **Team Reflection**: Did all members get a chance to share concerns and ideas? If not, revisit steps to incorporate everyone's perspective.

2.13 Common Pitfalls and Key Insights

- **Underestimating Bandwidth and Latency Needs**: Match protocols (e.g., Wi-Fi, LoRaWAN) with data volume and real-time needs.
- **Ignoring Interoperability**: Plan how different devices and services communicate, especially for scaling or adding features.
- **Overlooking Edge Cases**: Consider large crowds, device failures, or poor network coverage.
- **Security as an Afterthought**: Even basic security measures early can prevent major issues later.
- **Minimal Viable vs. Future-Proofing**: Start lean—add complexities only if there's clear user benefit, but anticipate possible growth or redundancy needs.

2.14 Conclusion

Workshop 2 built on the user-centric foundation of Workshop 1 by guiding you into the **technical, system-level planning** of IoT solutions. Through hands-on use of the *IoT Service Kit*—complete with maps, tokens, and cards—you refined your initial concepts to address key factors like **scalability**, **network reliability**, and **security**.

Remember that each design choice also relates to **non-functional requirements** (performance, fault tolerance, maintainability). As you continue to iterate, test, and reflect on your architectural choices, each new insight will strengthen your ability to **evaluate trade-offs** and **optimize communication flows**. With this expanded toolkit for planning and architecting IoT systems, you're now well-prepared to tackle more advanced challenges in future modules—whether that involves hardware prototyping, deeper analytics, or even exploring monetization models if relevant.

ONBOARDING INTERACTION
For first-time users, how will be the first interaction with the product?
What activities needs to be done?
WHAT IF THE PRODUCT
STANDALONE VS PARASITE
Is the product standalone or a parasite?
WHAT IF YOU

3. Internet of Things Product Development

3.1 Introduction and Preparation

Welcome to **Workshop 3: Internet of Things Product Development**. Building on your experiences from *Workshop 1* (user-centric ideation) and *Workshop 2* (system-level planning), this session guides you through **identifying problems, brainstorming solutions, and refining practical IoT concepts** in a collaborative environment.

You can approach this workshop in two ways:

1. *Develop Further*: If you completed Workshop 1 and 2, bring *your existing user scenarios, architecture concepts, or constraints* and refine those ideas by focusing on product development layers (e.g., user experience, market viability).
2. *Fresh Start*: Alternatively, if you are joining directly, you can create a new IoT concept in this session, using the same iterative steps described here.

Either path will benefit from an iterative, user-centric approach. Like the previous workshops, there is **no formal evaluation**; instead, you will collaborate to evolve your understanding of IoT product design and share insights through group activities.

Recommended Group Roles: We suggest designating a *facilitator* to keep discussions on track, a *note-taker* to capture ideas and decisions, and a *timekeeper* to ensure each step stays within the recommended durations.

Workshop Goals

By the end of this session, you should be able to:

- **Identify and Frame Problems**: Recognize user needs and real-world challenges suitable for IoT interventions.
- **Analyze Existing IoT Products**: Examine current smart products to learn from their features, pitfalls, and opportunities for improvement.
- **Generate New Ideas *Iteratively***: Use the *Mapping the IoT Toolkit* (or similar resources) to foster diverse design possibilities.
- **Refine and Enhance Designs**: Align each development step with real user feedback, technical feasibility, and market considerations.

Pre-Workshop Checklist

Before starting Workshop 3, ensure you have:

- **The Mapping the IoT Toolkit** (physical or digital): This includes *Activity Guides* and a *Deck* of thematic cards (Strategy, User & Context, Design, Interaction, Technology, Experience, Meaning) to help frame problems, analyze products, brainstorm, and refine your IoT concept.
- **A Collaborative Workspace**: Use a whiteboard, sticky notes, or an online board (e.g., Miro) for group discussions, sketching ideas, and documenting feedback.
- **Sample Project or User Story (Optional)**: Bring at least one real-world scenario—e.g., a wearable for diabetic patients or a smart home device for energy savings—that can serve as an illustrative example.

3.2 Objectives

- **Identify and Frame Problems** by studying user contexts and challenges, linking them to feasible IoT interventions.
- **Analyze Existing IoT Products** to learn from strengths, weaknesses, and feature gaps.
- **Brainstorm a Variety of Ideas** using structured methods, pushing beyond obvious solutions to innovative concepts.
- **Refine and Enhance** your chosen IoT concept by ensuring it meets user needs, technical feasibility, and potential market viability.

3.3 Learning Outcomes

By completing this workshop, you should be able to:

- **Apply Activity-Based Learning** to move systematically from project initiation to final refinement (with the option to revisit earlier steps if new insights arise).
- **Develop Storyboards and Elevator Pitches** to communicate your core value proposition and gather user feedback quickly.
- **Critically Evaluate Existing Products** and extract actionable lessons for your own IoT designs.
- **Use Structured Toolkits and Cards** (e.g., *Analysis Cards*, *Feature Map*, *Deck Cards*) for disciplined ideation and problem-solving.
- **Collaborate in Teams** by defining roles, exchanging feedback, and combining individual strengths effectively.
- **Prototype Quickly and Iteratively** (e.g., low-fidelity mockups, short user tests) to validate feasibility and user acceptance early on.

3.4 Related Lessons

Although not mandatory, familiarity with the following lessons (📺) will enrich your understanding:

- **Lesson 7 – Human Factors and Interaction**: Explores user-centered design principles, accessibility, and ergonomics.
- **Lesson 8 – Design Strategies and Prototyping**: Covers rapid prototyping methods, iterative testing, and design validation.

3.5 Workshop Material Access

Resources **Choose your resources:**

- **Digital Materials**: Access the Miro Board from .
  - *Note*: Free Miro plans may restrict certain features. Plan your board layout or consider alternative digital tools if needed.
- **Physical Materials**: Download and print the card sets from .

Credits to the original creators: Mapping The IoT Toolkit

3.6 Recommended Outcomes of the Workshop

Depending on which of the four core activities you focus on, this workshop can yield different deliverables. Ideally, by the end of the session, each group will produce **at least one** of the following outcomes:

- **Project Brief (for "I Want to Start a New Project")**:
  - A concise document framing the IoT challenge, user needs, and initial technology ideas.
  - Highlights any unique angles or value propositions discovered during your early exploration.
- **Comparative Analysis (for "I Want to Analyze Smart Products")**:
  - A structured review of existing IoT products, using tools like *Analysis Cards* and a *Feature Map*.
  - Identifies common strengths, weaknesses, and opportunities—valuable for guiding your own product design.
- **Ideation Set (for "I Want to Brainstorm Ideas")**:
  - A wide-ranging collection of raw concepts, sketches, or "What If" scenarios generated with the *Deck Cards*.
  - May include quick prototypes, rough user flows, or themes for further exploration in later activities.
- **Refined IoT Concept (for "I Want to Make My Concept Better")**:
  - A more polished solution incorporating feedback from peers and addressing technical, user-experience, or strategic gaps.
  - May include a **Storyboard** illustrating user flow, a 3-minute **Elevator Pitch** summarizing core benefits, and **Reflection Notes** documenting key decisions or remaining questions.

Measuring Success

Success Measures: As you reach the end of this workshop, consider:

- Have you clearly documented *which network protocol(s)* are used and *why*?
- Does your architecture diagram illustrate **data flow** from sensors to services, including any **edge** or **cloud** components?
- Have you listed at least **one security safeguard** (e.g., encryption, authentication) relevant to your environment?
- Did you address **user constraints** or environment-specific limitations (e.g., limited bandwidth, intermittent connectivity)?
- *Have you sought input or feedback* from peers or stakeholders to catch gaps early?

3.7 Quick-Reference Table

Table 3.1: Workshop 3 Steps, Activities, Time, and Outputs (detailed)

Step	Activity	Time	Main Output
1	*I Want to Start a New Project*	15 min	Project brief framing the problem and initial tech selections
2	*I Want to Analyze Smart Products*	15 min	Comparative review or feature map of existing IoT products
3	*I Want to Brainstorm Ideas*	15–20 min	Diverse set of raw concepts generated from the *Deck Cards*
4	*I Want to Make My Concept Better*	15–20 min	Refined IoT solution (storyboard, key enhancements, next steps)

Note on Iteration: You need not follow these four activities linearly. For instance, after analyzing

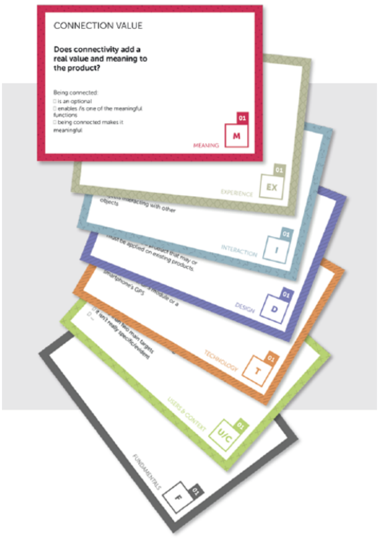
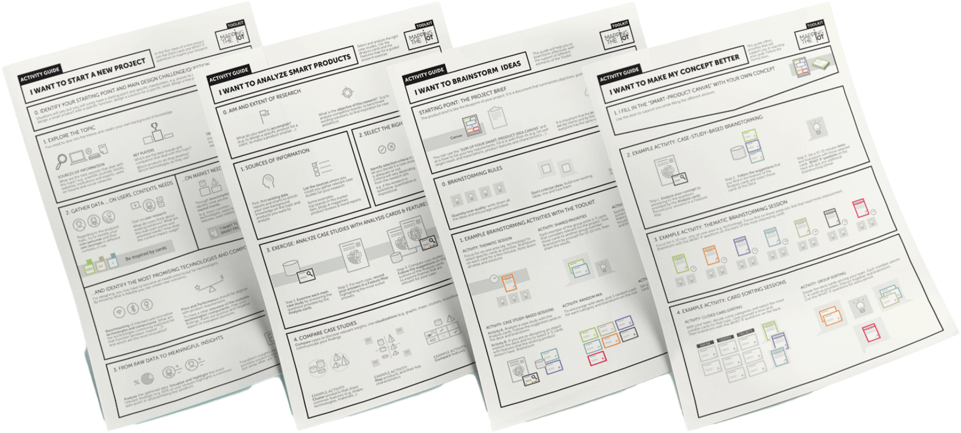
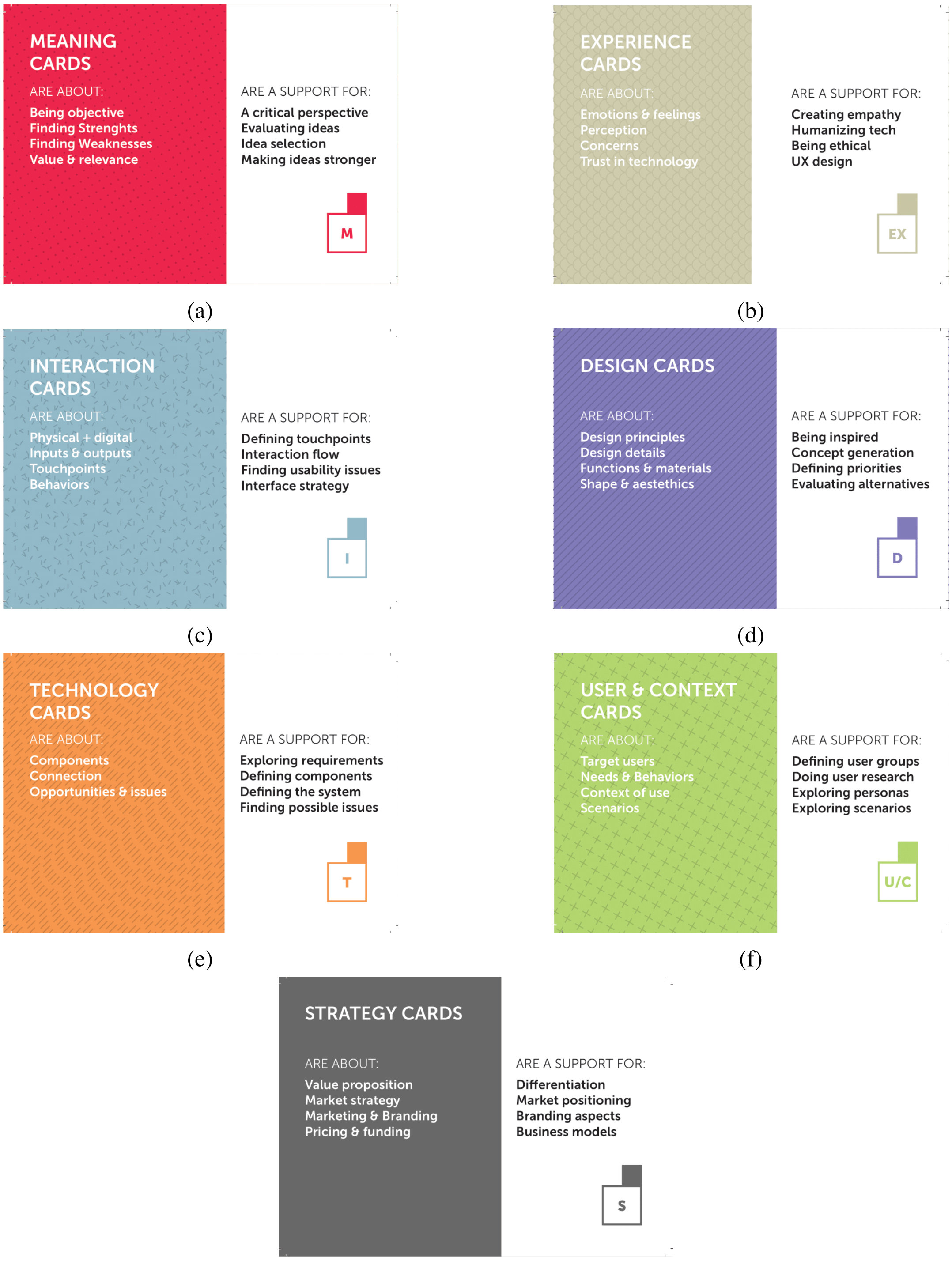

CONNECTION VALUE
Does connectivity add a real value and meaning to the product?
Being connected
□ is an optional
□ enables /is one of the meaningful functions
□ being connected makes it meaningful
MEANING
01
M
EXPERIENCE
EX
INTERACTION
I
DESIGN
D
TECHNOLOGY
T
USER & CONTEXT
U/C
FUNDAMENTALS
F

Figure 3.1: A close-up of the workshop's thematic Deck. Each category—Interaction, Design, Technology, User & Context, Fundamentals, Meaning, Experience, Strategy—corresponds to a different facet of IoT product development. By combining multiple card types, teams are encouraged to consider everything from core functionality and user interactions to branding and vision.


ACTIVITY GUIDE
I WANT TO START A NEW PROJECT
ACTIVITY GUIDE
I WANT TO ANALYZE SMART PRODUCTS
ACTIVITY GUIDE
I WANT TO BRAINSTORM IDEAS
ACTIVITY GUIDE
I WANT TO MAKE MY CONCEPT BETTER

Oh

Figure 3.2: The *Activity Guides* form a core element of the Toolkit, offering structured instructions to support each of the four main activities: "I want to start a new project," "I want to analyze smart products," "I want to brainstorm ideas," and "I want to make my concept better."



MEANING CARDS

ARE ABOUT:

Being objective
Finding Strenghts
Finding Weaknesses
Value & relevance

ARE A SUPPORT FOR:

A critical perspective
Evaluating ideas
Idea selection
Making ideas stronger

M

(a)

EXPERIENCE CARDS

ARE ABOUT:

Emotions & feelings
Perception
Concerns
Trust in technology

ARE A SUPPORT FOR:

Creating empathy
Humanizing tech
Being ethical
UX design

EX

(b)

INTERACTION CARDS

ARE ABOUT:

Physical + digital
Inputs & outputs
Touchpoints
Behaviors

ARE A SUPPORT FOR:

Defining touchpoints
Interaction flow
Finding usability issues
Interface strategy

I

(c)

DESIGN CARDS

ARE ABOUT:

Design principles
Design details
Functions & materials
Shape & aesthetics

ARE A SUPPORT FOR:

Being inspired
Concept generation
Defining priorities
Evaluating alternatives

D

(d)

TECHNOLOGY CARDS

ARE ABOUT:

Components
Connection
Opportunities & issues

ARE A SUPPORT FOR:

Exploring requirements
Defining components
Defining the system
Finding possible issues

T

(e)

USER & CONTEXT CARDS

ARE ABOUT:

Target users
Needs & Behaviors
Context of use
Scenarios

ARE A SUPPORT FOR:

Defining user groups
Doing user research
Exploring personas
Exploring scenarios

U/C

(f)

STRATEGY CARDS

ARE ABOUT:

Value proposition
Market strategy
Marketing & Branding
Pricing & funding

ARE A SUPPORT FOR:

Differentiation
Market positioning
Branding aspects
Business models

S

(g)

Figure 3.3: (a) "Meaning Cards" prompt a deeper assessment of genuine value. (b) "Experience Cards" focus on user emotions, trust, and engagement. (c) "Interaction Cards" illustrate how users operate or receive feedback. (d) "Design Cards" address aesthetics, form factors, and frameworks. (e) "Technology Cards" define feasibility and constraints. (f) "User & Context Cards" explore target audiences and usage patterns. (g) "Strategy Cards" outline long-term objectives, branding, and market positioning.

existing products (Step 2), you might revisit Step 1 to reframe your user needs or constraints if new insights arise.

3.8 Step-by-Step Guide

Below is a detailed guide to each of the four main activities. Align these steps with the *Quick-Reference Table* (Table 3.1) as you progress. **Facilitator Tip**: For each activity, consider how to incorporate quick user checks or stakeholder feedback loops.

3.8.1 Activity 1: I Want to Start a New Project

In this activity, you **frame the problem context** for a brand-new IoT idea. You'll use thematic cards (e.g., *User & Context*, *Meaning*) to clarify user needs, define goals, and ground your project in a meaningful challenge.

Step 1: Identify Your Main Design Challenge

- **Constraints and Stakeholders**: Consider any restrictions related to available funding, the target demographic's age and abilities, or environmental factors like climate and terrain. For example, assess whether the design requires high power consumption, which could pose challenges in remote areas, or if it needs to cater to users with limited technical proficiency.
- **Examples**: Examples could include projects like developing a solar-powered device tailored for rural families seeking sustainable energy solutions or creating a smart city tool designed to streamline urban infrastructure monitoring, such as adaptive traffic lights or public safety sensors.

Step 2: Explore the Topic

- **Research Sources**: Tech blogs, patents, or scientific journals on IoT trends.
- **Key Players**: Identify competing products or relevant ecosystem partners (e.g., local city data).
- **User & Context Cards**: See Figure 3.3(f) for ways to probe user needs or constraints.

Step 3: Gather Data on Users, Contexts, and Needs

- **User Personas**: Conduct quick interviews (if possible) or hypothesize typical users to understand daily routines or pain points.
- **Market Differentiation**: Reflect on how your idea stands out—cheaper sensors, simpler interfaces, new functionalities, or business viability.
- **Experience Cards**: Refer to Figure 3.3(b) for emotional or ethical factors that might influence adoption.

Step 4: Identify Promising Tech Components

- **Technology Cards**: Shown in Figure 3.3(e), these help you evaluate sensor types, connectivity, or data analytics.
- **Meaning Cards**: See Figure 3.3(a) to align chosen tech with your project's overarching value (e.g., sustainability or health).

Step 5: Turn Raw Data into Insights

- **Design Cards**: Figure 3.3(d) can guide decisions about form factors or user flow.
- **Strategy Cards**: Figure 3.3(g) can spark discussion on branding or cost model if you plan to monetize.

Progress Check: Before finalizing your project brief, ask: "Did we define user motivations clearly? Have we considered cost, environment, or security constraints?"
Outcome of This Activity: A **project brief or mini-proposal**, describing the problem, user

context, initial technology choices, and the unique value proposition of your idea. Remember to check if users or stakeholders might pay for this product or if it's purely conceptual.

3.8.2 Activity 2: I Want to Analyze Smart Products

Here, you **examine existing IoT products** to understand pitfalls, strong features, and user expectations. This helps ensure your solution doesn't repeat common mistakes. You'll use *Analysis Cards* and a *Feature Map* to organize insights.

Step 1: Define Research Scope

- **Example Goal**: "Identify best practices in wearable health devices" or "Evaluate connectivity solutions in smart agriculture tools."
- **Agenda**: Decide how many products to analyze and how in-depth you will go (basic feature scan vs. thorough teardown).

Step 2: Collect Background Information

- **Sources**: Tech websites, user forums, product reviews, academic studies, or real user feedback.
- **Analysis Cards**: Sort products by type, target market, cost, or any other relevant category.

Step 3: Select Relevant Case Studies

- **Criteria**: Product typology, target user group, or innovation level.
- **Filtering**: Narrow down to items that share similar goals or constraints to your own project. E.g., "Low-power wearables" or "Voice-activated devices."

Step 4: Fill Out the Feature Map

- **Product Description**: Summarize functions, user type, and any subscription model.
- **Strengths & Weaknesses**: Note cost, design, connectivity, user experience, security measures.
- **Data and Interaction**: Observe how data is collected, processed, and visualized (see *Interaction Cards*, Figure 3.3(c)).

Step 5: Compare and Extract Insights

- **Look for Patterns**: Do products fail in offline scenarios? Are they overbuilt or too pricey for casual users?
- **Actionable Lessons**: Identify repeated pitfalls or missed opportunities to address in your own design.
- **Real-World Example**: Mention a known IoT recall or success story to illustrate how certain flaws or strengths played out.

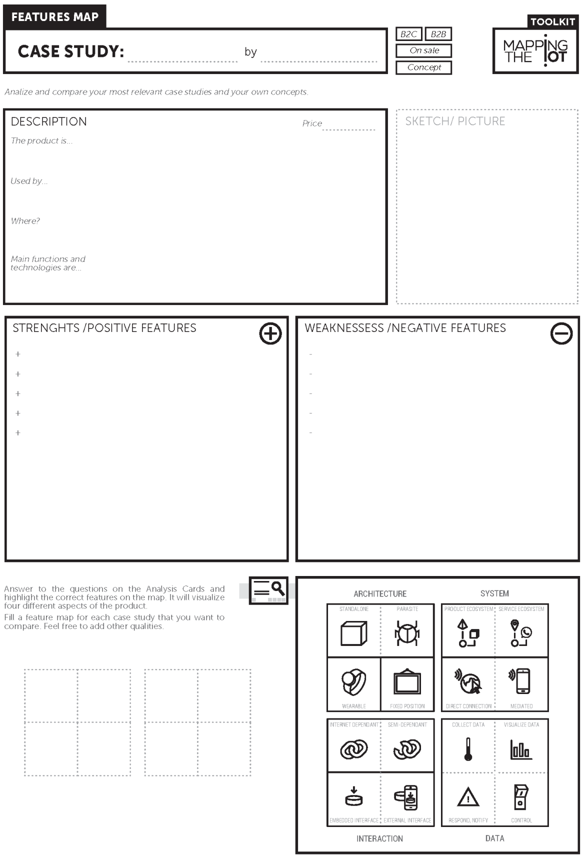
Progress Check: After your comparative review, ask: "Which features consistently impressed or disappointed us? Which enhancements or pitfalls can inform our design decisions?"
Outcome of This Activity: **Analysis Summaries** for each product, plus a **comparative review** highlighting opportunities or gaps—key when deciding which features to adopt or avoid.

3.8.3 Activity 3: I Want to Brainstorm Ideas

Having explored potential problems (subsection 3.8.1) and studied existing solutions (subsection 3.8.2), you're ready to **rapidly generate creative ideas**. This activity uses the *Deck Cards* (Figures 3.3) to stimulate out-of-the-box thinking.

Step 1: Brainstorming Rules

- **Quantity Over Quality**: Encourage all ideas—no immediate feasibility checks.
- **Time-Box Sessions**: Maintain momentum with short bursts (5–10 minutes).



FEATURES MAP

CASE STUDY: .... by ....

B2C B2B
On sale
Concept

TOOLKIT
MAPPING THE IOT

Analize and compare your most relevant case studies and your own concepts.

DESCRIPTION — *Price* ....

The product is...

Used by...

Where?

Main functions and technologies are...

SKETCH/ PICTURE

STRENGHTS /POSITIVE FEATURES ⊕

+
+
+
+
+

WEAKNESSESS /NEGATIVE FEATURES ⊖

-
-
-
-
-

Answer to the questions on the Analysis Cards and highlight the correct features on the map. It will visualize four different aspects of the product.

Fill a feature map for each case study that you want to compare. Feel free to add other qualities.

Figure 3.4: Features Map

- **Mix and Fuse Ideas**: Combine earlier suggestions or surprising card pairs. “Wild ideas” can spark innovation.

Step 2: Set Your Focus

- **Project Brief Check-In**: Revisit your project constraints or product analyses to keep brainstorming grounded.
- **Choose a Brainstorm Style**: *Thematic*, *Shared priorities*, *Random mix*, or *Case study-based*.

Step 3: Explore Possible Methods

- **Thematic Session**: Focus on one card category (e.g., *Technology*), generating multiple ideas before clustering them.
- **Shared Priorities**: Each member picks 1–3 crucial cards, brainstorms individually, then swaps.
- **Random Mix**: Draw from multiple categories (e.g., *Meaning*, *Experience*, *Design*) and merge them into a single concept.
- **Case Study-Based Session**: Revisit your analysis from subsection 3.8.2, focusing on 2–5 features to improve.

Step 4: Refine or Cluster Ideas

- **Group Similar Concepts**: Create clusters (e.g., sustainability, advanced analytics).
- **Prioritize for Next Steps**: Mark top ideas to refine in subsection 3.8.4.

Progress Check: Before wrapping up, ask: “Do we have at least one ‘wild idea’ that could push

boundaries? Did we identify trade-offs like cost or complexity among our favorites?"
Outcome of This Activity: A **diverse set of raw ideas**—likely more than you'll fully develop. Use these as the foundation for final refinement, user testing, or further iteration.

3.8.4 Activity 4: I Want to Make My Concept Better

In this final step, you **refine and enhance** your IoT concept, revisiting both *Analysis Cards* and *Deck Cards* for fresh insights. Consider quick prototypes or user tests if feasible.

Step 1: Fill in the Smart Product Canvas

- **Contextualize Your Idea**: Summarize user needs, environment constraints, core goals and fill the Smart Product Canvas shown in 3.5.
- **Specify Key Components**: Note *Technologies*, *User & Context Requirements*, *Security*, and *Interactions* explicitly.
- **Highlight Value and Differentiation**: In "Connectivity & Meaning," articulate why your solution stands out—cost, usability, brand potential, etc.

Step 2: Use Activity Cards to Identify Gaps

- **Analyze Your Concept**: Check *Analysis Cards* for weaknesses (unmet user needs, security flaws, offline mode).
- **Suggest Improvements**: Link each gap to relevant *Deck Cards* (Figure 3.1). For usability issues, consult *Interaction* or *Experience* cards.
- **Incorporate Security/Privacy**: If your device handles sensitive data, consider encryption, local data storage, or user consent flows.

Step 3: Thematic Brainstorming or Card Sorting

- **Closed Card Sorting**: Label improvements as "Must-Have," "Needs Discussion," "Future Feature."
- **Group Sorting**: Each member picks 2–4 improvement cards, merges them into an updated concept.
- **Time-Boxed Brainstorm**: Spend 10–15 minutes refining each category of improvements.

Step 4: Document and Validate

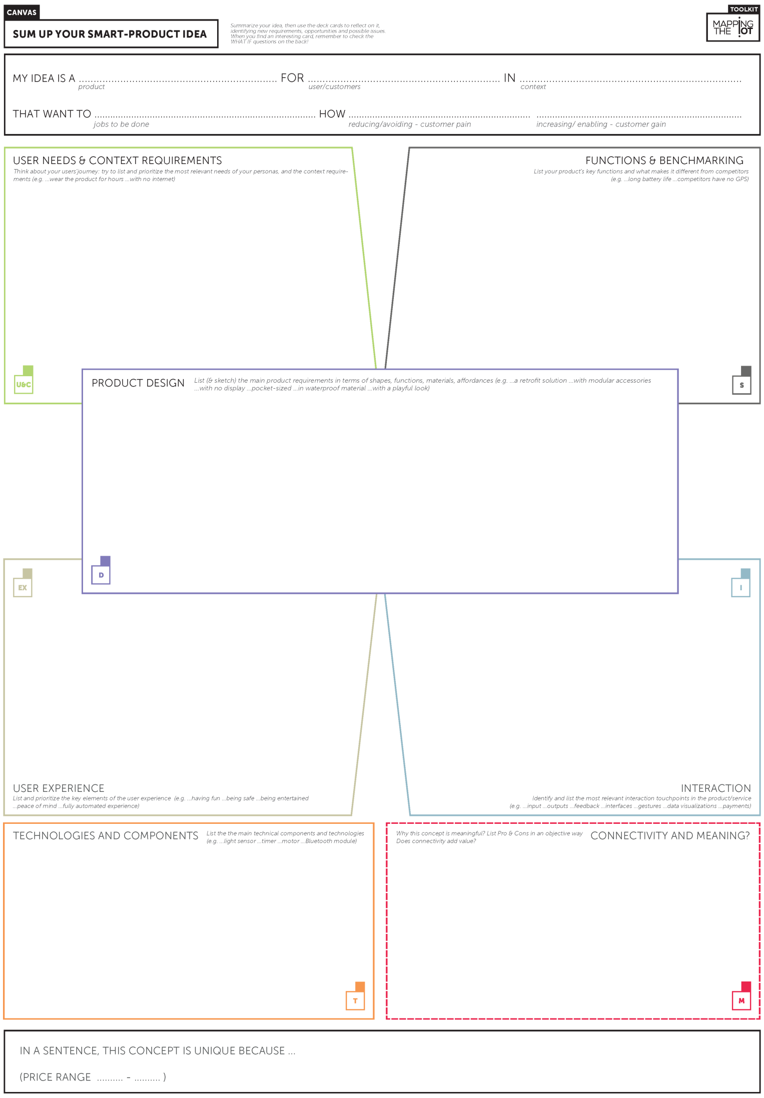
- **Update the Canvas and Prototype**: Adjust relevant sections or create a paper/digital mockup.
- **User Feedback**: If possible, present to peers or real users. Validate if changes solve actual pain points.
- **Check Alignment**: Confirm refinements still meet objectives from subsection 3.8.1.

Outcome of This Activity: A more **robust IoT concept** addressing previous gaps. You may have:

- **A Refined Smart-Product Canvas**: Documenting new insights and design directions.
- **A Low-Fidelity Prototype**: Illustrating how the concept works in practice (e.g., cardboard mockups, simple sensor demos).
- **An Updated Elevator Pitch**: Summarizing core value, feasibility, and benefits in about three minutes.

3.9 Common Pitfalls and Key Insights

- **Skipping User Research**: Quick persona or scenario checks can prevent building irrelevant solutions.
- **Overloading Features**: Resist the urge to add everything. Start lean; iterate based on user feedback.



CANVAS

SUM UP YOUR SMART-PRODUCT IDEA

Summarize your idea, then use the deck cards to reflect on it, identifying new requirements, opportunities and possible issues. When you find an interesting card, remember to check the WHAT IF questions on the back!

TOOLKIT

MAPPING THE IOT

MY IDEA IS A .... (*product*) FOR .... (*user/customers*) IN .... (*context*)

THAT WANT TO .... (*jobs to be done*) HOW .... (*reducing/avoiding - customer pain*) .... (*increasing/ enabling - customer gain*)

USER NEEDS & CONTEXT REQUIREMENTS

Think about your users'journey: try to list and prioritize the most relevant needs of your personas, and the context requirements (e.g. ...wear the product for hours ...with no internet)

FUNCTIONS & BENCHMARKING

List your product's key functions and what makes it different from competitors (e.g. ...long battery life ...competitors have no GPS)

PRODUCT DESIGN

List (& sketch) the main product requirements in terms of shapes, functions, materials, affordances (e.g. ...a retrofit solution ...with modular accessories ...with no display ...pocket-sized ...in waterproof material ...with a playful look)

USER EXPERIENCE

List and prioritize the key elements of the user experience (e.g. ...having fun ...being safe ...being entertained ...peace of mind ...fully automated experience)

INTERACTION

Identify and list the most relevant interaction touchpoints in the product/service (e.g. ...input ...outputs ...feedback ...interfaces ...gestures ...data visualizations ...payments)

TECHNOLOGIES AND COMPONENTS

List the the main technical components and technologies (e.g. ...light sensor ...timer ...motor ...Bluetooth module)

CONNECTIVITY AND MEANING?

Why this concept is meaningful? List Pro & Cons in an objective way Does connectivity add value?

IN A SENTENCE, THIS CONCEPT IS UNIQUE BECAUSE ...

(PRICE RANGE .... - ....)

Figure 3.5: The *Smart Product Canvas* lets you summarize key aspects of your IoT solution, from user needs and context to product design, technology, and meaning. Filling it out after each iteration helps you track progress and maintain a user-centered focus.

- **Neglecting Security**: Demonstrate how easily unprotected data can be compromised. Bake in security from the start.
- **Forgetting Offline Scenarios**: Plan fallback modes or local data storage for limited connectivity.
- **Underestimating Market Factors**: Even a great technical design can fail without a cost model or marketing strategy.
- **Insufficient Iteration**: Rushing to finalize without repeated feedback loops can leave critical usability or feasibility issues.
- **Misaligned Value Proposition**: Double-check that your solution solves a real user problem worth paying (time or money) for.

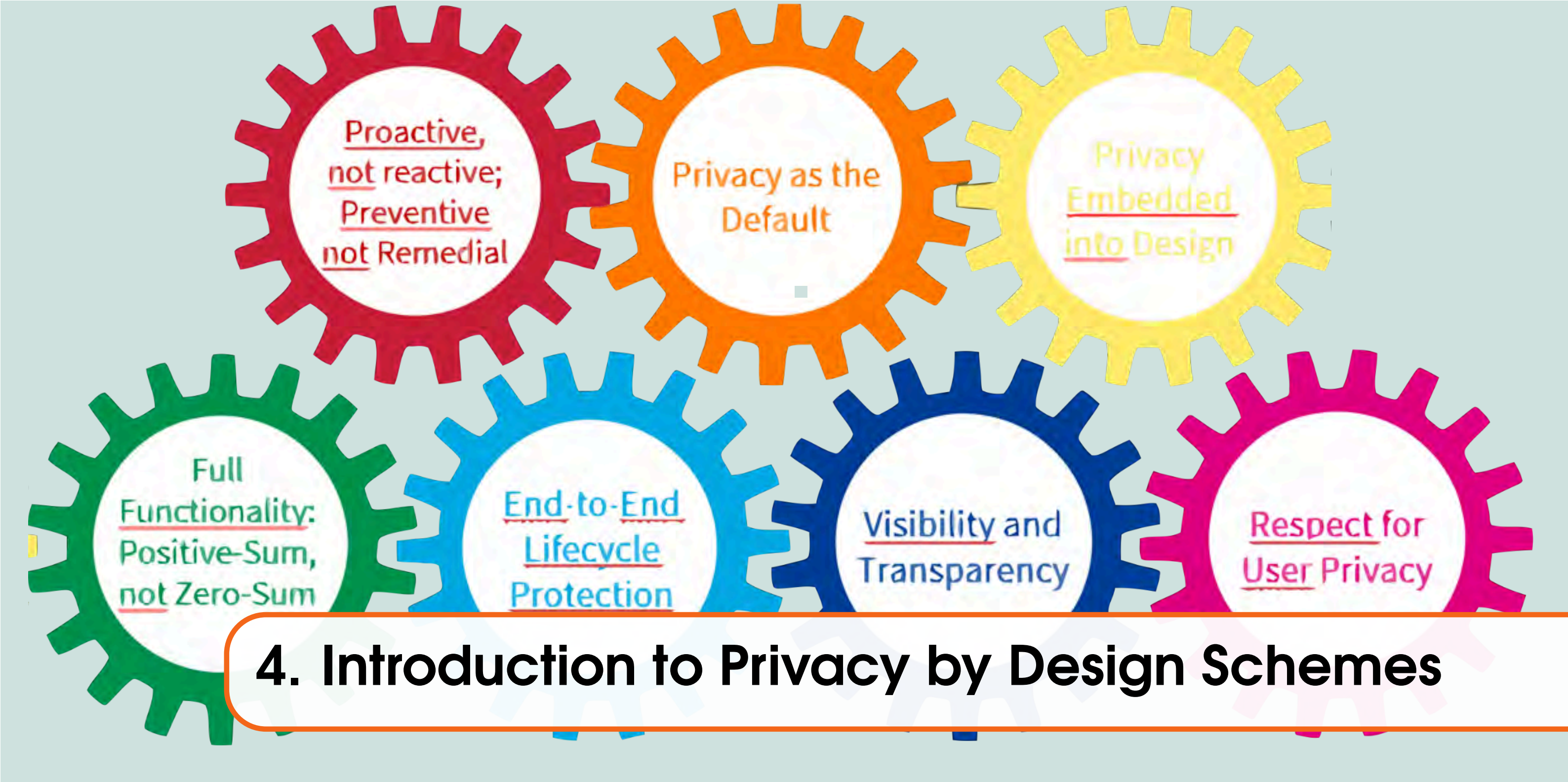
3.10 Reflection and Team Retrospective

- **Team Retro**: Spend a few minutes discussing what each member learned, which activities felt most valuable, and what can be improved in future workshops.
- **Collaboration vs. Individual Work**: If you worked solo, consider where group input or external feedback could have helped. If you worked in a team, did all voices contribute equally?

3.11 Conclusion

Workshop 3 has guided you through the **full lifecycle of IoT product design**, from discovering user needs to refining a near-finished concept. By carrying out the four core activities—*Starting a New Project, Analyzing Existing Products, Brainstorming Ideas*, and *Enhancing Your Concept*—you have built on the creative groundwork of Workshop 1 and the technical planning skills developed in Workshop 2.

Throughout these steps, you have strengthened your ability to **balance design ideals with practical constraints**, ensuring your prototype remains user-centric, technically viable, and market-aware. You are now equipped to **transform your concept into tangible outcomes**, whether that means building a minimum viable product (MVP), creating a compelling pitch for stakeholders, or moving on to more advanced modules (e.g., *Privacy by Design*, *Hardware Prototyping*, or *Scaling to Production*). Ultimately, Workshop 3 positions you to drive meaningful innovation in the IoT domain, shaping solutions that truly **resonate with users** and broader societal needs.


Proactive, not reactive; Preventive not Remedial
Privacy as the Default
Privacy Embedded into Design
Full Functionality: Positive-Sum, not Zero-Sum
End-to-End Lifecycle Protection
Visibility and Transparency
Respect for User Privacy

4. Introduction to Privacy by Design Schemes

4.1 Introduction and Preparation

Welcome to **Workshop 4: Introduction to Privacy by Design Schemes**. As IoT solutions collect, store, and transmit large volumes of data—often containing sensitive user information—it is critical to embed privacy safeguards **from the very beginning** of the design process. This workshop builds on foundational lessons from Workshops 1, 2, and 3, guiding you to identify privacy risks, propose mitigation strategies, and design IoT systems that respect user data and autonomy.

You can approach this workshop in two ways:

1. *Develop Further*: If you completed Workshops 1, 2, or 3, bring any *existing IoT concepts, storyboards, or architecture diagrams* and focus on weaving in strong privacy measures.
2. *Fresh Start*: Alternatively, if you are joining directly, you can create a new IoT project idea in this session, using the same systematic steps below.

Recommended Group Roles: We encourage appointing a *facilitator* (to keep discussions on track), a *note-taker* (to capture privacy-related decisions), and a *timekeeper* (to manage pacing). This ensures balanced participation and consistent documentation of privacy considerations.

Workshop Goals

By the end of this workshop, you should be able to:

- **Understand Privacy by Design (PbD)**: Learn how to integrate privacy principles across the entire IoT lifecycle, from ideation to deployment.
- **Identify Privacy Risks**: Examine which parts of an IoT system are most vulnerable to data misuse or exposure.
- **Explore Mitigation Strategies**: Discover frameworks, patterns, and design techniques that minimize privacy threats.
- **Implement Ethical and Legal Responsibilities**: Ensure your IoT designs align with user expectations, **relevant regulations**, and fundamental privacy rights.

Pre-Workshop Checklist

To make the most of Workshop 4, please have:

- **The Privacy Cards**: Physical or digital resources outlining privacy frameworks, principles, and design patterns. These will be used to brainstorm and implement privacy measures.
- A **collaborative workspace** (whiteboard, sticky notes, or an online board) for ideation.
- **Sketching materials** (pens, markers, blank paper) or a digital drawing platform.
- **Optional Real-World Example**: If you have a partially designed IoT system (from prior workshops) or an existing IoT product's data flow, bring it to anchor your privacy discussions in a concrete scenario.

4.2 Objectives

- **Embed Privacy Principles** into your IoT designs to proactively address data protection and user autonomy.
- **Identify Potential Privacy Vulnerabilities** and explore recognized frameworks or patterns to mitigate them.
- **Reflect on Ethical and Legal Responsibilities** of IoT developers, ensuring **real-world compliance** with user expectations and relevant regulations.
- **Integrate User-Centric Privacy Controls** that enhance trust, transparency, and user control over personal data.

4.3 Learning Outcomes

By completing Workshop 4, you should be able to:

- **Discuss Major Privacy Frameworks** (e.g., ISO/IEC 29100, Cavoukian's Principles, Hoepman's Strategies) and understand their relevance to IoT.
- **Apply Privacy Patterns** (e.g., Data Minimization, Location Granularity) in practical IoT scenarios.
- **Analyze and Redesign** an existing IoT system to integrate privacy-by-design principles, acknowledging **trade-offs** with cost or usability.
- **Facilitate Privacy-Focused Brainstorming** sessions using a *Privacy Deck* or similar toolkit.
- **Communicate Privacy Features** effectively to non-technical stakeholders and end users, highlighting compliance and user benefits.

4.4 Related Lessons

Although not mandatory, familiarity with the following lessons (📺) will enrich your understanding:

- **Lesson 6 – Privacy and Security**: Covers fundamental data protection concepts, threat models, and risk assessment in IoT contexts.

4.5 Workshop Material Access

Resources **Choose your resources:**

- **Digital Materials**: Use the Miro board for interactive sessions at 🟨.
- **Physical Materials**: Download and print relevant guides from ◆.

Warning: Unlike the other workshops, this one does not rely on a single source of material. Instead, we have developed a collection of privacy cards (or "privacy decks") inspired by well-known privacy-by-design frameworks (see Figure 4.2). Think of the privacy cards as a flexible tool: adapt them to best explore privacy considerations for your chosen IoT system. Through experimentation and discussion, they can spark new ideas and guide you toward designing more privacy-conscious solutions. ■

4.6 Recommended Outcomes of the Workshop

By the end of this workshop, each group should ideally produce:

- **A Privacy-by-Design Enhanced Concept**: Updated IoT system diagrams or storyboards reflecting newly implemented privacy measures.
- **Risk Assessment Notes**: Listing each identified privacy vulnerability and the mitigation strategies applied.
- **Brief Presentation or Elevator Pitch**: Demonstrating how the design complies with privacy principles and why it's trustworthy.

Measuring Success

Success Measures: As you reach the end of this workshop, consider:

- Have you *identified and prioritized* all major **privacy risks** in your IoT system?
- Did you **minimize unnecessary data collection** or adjust data granularity (e.g., location precision) to reduce privacy threats?
- Are there **clear, user-friendly controls** for managing data sharing, consent, and permissions?
- Have you **implemented at least one privacy safeguard** (e.g., anonymization, encrypted communication) relevant to your design?
- Are your **compliance and ethical considerations** (e.g., GDPR, local regulations) addressed early on, rather than retrofitted?
- *Have you sought feedback* from peers or stakeholders to validate that your privacy measures make sense in real-world usage?

■

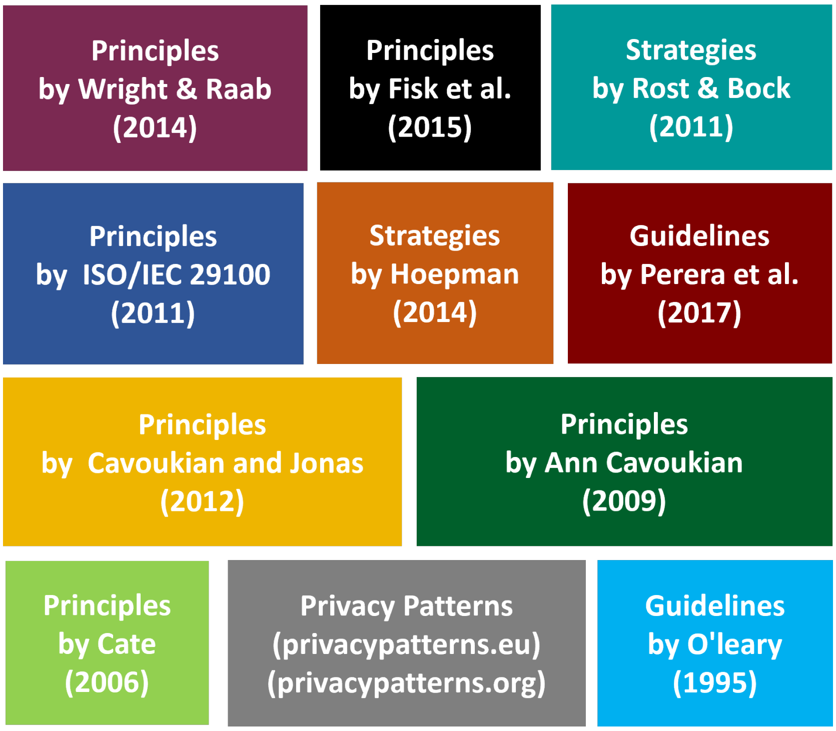
4.7 Quick-Reference Table

Table 4.1 outlines each main step, a recommended duration, and the primary output. Note that this workshop is iterative: if new privacy vulnerabilities arise mid-way, you may return to an earlier step.

Table 4.1: Workshop 4 Steps, Activities, Time, and Outputs (Detailed)

Step	Activity	Time	Main Output
1	Introduction to Privacy Frameworks	10–15 min	Short-list of relevant privacy principles or frameworks
2	Identifying Privacy Risks	15–20 min	Risk map or annotated diagram of vulnerabilities
3	Privacy Deck Activity	25–30 min	Revised design notes/storyboards integrating privacy principles
4	Refining Your Design + Pitch	15–20 min	Polished concept + 3-min pitch highlighting privacy enhancements

Figure 4.1: These ideation cards illustrate how regulatory principles like data protection and privacy can be embedded early in the design process, offering a structured way for designers to identify potential risks and align solutions with emerging legal frameworks. By highlighting moral and social dimensions alongside legal obligations, the cards encourage a shift from compliance-based thinking to more creative, user-centered considerations. Each card presents tangible prompts, inspiring designers to explore opportunities for privacy-preserving features without stifling innovation. Whether used in workshops or team sessions, they foster collaborative thinking that integrates social, technical, and legal perspectives into the heart of IoT system design. These cards equip practitioners to create responsible technologies that balance user needs with regulatory requirements. Luget et al (2015). `https://doi.org/10.1145/2702123.2702142`



Principles by Wright & Raab (2014)

Principles by Fisk et al. (2015)

Strategies by Rost & Bock (2011)

Principles by ISO/IEC 29100 (2011)

Strategies by Hoepman (2014)

Guidelines by Perera et al. (2017)

Principles by Cavoukian and Jonas (2012)

Principles by Ann Cavoukian (2009)

Principles by Cate (2006)

Privacy Patterns (privacypatterns.eu) (privacypatterns.org)

Guidelines by O'leary (1995)

Figure 4.2: An overview of Privacy-by-Design (PbD) Schemes. These high-level frameworks offer principles and strategies to ensure data protection, minimize unnecessary collection, and respect user autonomy throughout IoT system design.

4.8 Workshop Overview

In this workshop, you will:

- **Revisit IoT Concepts**: If available, bring your designs from previous workshops or real-world IoT products to identify privacy needs.
- **Explore Privacy Frameworks**: Familiarize yourself with privacy-by-design strategies, referencing Figures 4.2, 4.1, and the long table (Table 4.2) for deeper details.
- **Engage in a Privacy Deck Activity**: Use cards representing key privacy principles to guide design modifications (e.g., data minimization, user consent).
- **Demonstrate Findings**: Present your updated concept, focusing on how you tackled vulnerabilities and *why* it strengthens trust and compliance.

4.9 Step-by-Step Guide

Below is a structured approach to integrating privacy-by-design principles into your IoT concept. Align these steps with the **Quick-Reference Table** (Table 4.1). Feel free to adapt timings or revisit earlier steps if new privacy issues emerge.

Step 1: Introduction to Privacy Frameworks (10–15 Minutes)

1. **Form Small Groups or Work as a Whole Team:**
   - Assign roles (facilitator, note-taker). Clarify the scope of your IoT concept (size, environment, user type).
2. **Review Major Privacy Frameworks:**
   - Look at Figures 4.2 and 4.1, plus Table 4.2.
   - Encourage Teams to Filter: Focus on frameworks or principles most relevant to your user scenarios or data flows.
3. **Pick 2–3 Key Principles:**
   - E.g., “Minimise data” or “Offer user control” or “Encrypt by default.” These will shape your design moving forward.

Outcome: A short list of frameworks/principles tailored to your IoT concept, making the next steps more targeted.

> **Prompt:**
> - Are you focusing on compliance (e.g., GDPR) or user trust, or both?
> - Which frameworks best match your environment (home, city, industrial, healthcare)?

Step 2: Identifying Privacy Risks (15–20 Minutes)

1. **Examine Your IoT Diagram or Concept:**
   - Mark data-collection points (sensors, user inputs), data-storage locations (cloud, edge), and data-sharing mechanisms (APIs, third parties).
2. **Pinpoint Vulnerabilities:**
   - Identify unencrypted channels, excessive data collection, or unclear user consent flows.
   - Rate each vulnerability by impact (high/medium/low) and likelihood.
3. **Create a Risk Map:**
   - Visualize hotspots. This can be done via color coding or a digital board if remote.
   - Decide which vulnerabilities need immediate action (high impact) vs. those that can be addressed later.

Outcome: An annotated risk map or diagram clarifying which privacy threats are most critical to tackle first.

Questions:

- Does your system handle personal data like location, health info, or user credentials?
- Could third-party services be a weak link (e.g., external data analytics or payment gateways)?

■

Step 3: Privacy Deck Activity (25–30 Minutes)

1. **Review the Deck Cards:**
   - Each card focuses on a principle or pattern (e.g., "Data Minimization," "Pseudonymous Messaging," "Encrypted Communication").
2. **Form Subgroups or Remain as One Group:**
   - Discuss how each card might solve or reduce a specific vulnerability from Step 2.
   - Example: "Data Minimization" could address overly broad sensor data collection.
3. **Combine or Customize Principles:**
   - Some strategies are complementary, e.g., "Minimise + Hide," or "User Control + Encryption."
   - Acknowledge potential trade-offs (e.g., less data = simpler privacy but possibly reduced functionality).
4. **Document Proposed Changes:**
   - On sticky notes or in a shared doc, outline specific design shifts (e.g., removing detailed location data, adding end-to-end encryption).

Outcome: A refined **privacy-enhanced design plan**, listing concrete modifications or new features aligned with chosen privacy principles.

Real-World Reference:

- Recall any known IoT data breaches or compliance cases (e.g., unencrypted baby monitors). Reflect on how your design could avoid similar pitfalls.

■

Step 4: Refining Your Design + Pitch (15–20 Minutes)

1. **Consolidate Design Updates:**
   - Gather all your sticky notes or digital comments from Step 3.
   - Prioritize changes (must-do vs. nice-to-have).
2. **Address User Consent & Security Events:**
   - Clarify how users grant or revoke permissions.
   - Define procedures for potential data breaches or suspicious activities.
3. **Create a Brief "Before-and-After" Diagram:**
   - Show how your system looked pre-privacy enhancements and how it looks now.
   - Emphasize improved data flows, minimized collection, and stronger user control.
4. **Prepare a 3-Minute Elevator Pitch:**
   - Summarize top privacy challenges, your solutions, and user/market benefits.
   - Mention Next Steps: If you foresee further modules (e.g., Non-Functional Requirements), highlight how privacy measures feed into them.

Outcome: A polished IoT concept with integrated privacy solutions, plus a short pitch explaining the value and compliance aspects of your design.

Idea Check:

- Did you create easy-to-understand user settings or "privacy dashboards"?
- Have you tested any chosen solutions with real or proxy users to confirm usability?

■

4.10 Activity Reflection

Reflecting on your design process can help solidify lessons learned and highlight areas for improvement:

- **Data Minimization**: Did you reduce or eliminate unnecessary data points? How does this impact functionality?
- **User Autonomy**: Are users informed about data usage? Can they opt out or anonymize certain data streams?
- **Compliance Review**: Have you aligned with recognized standards (e.g., GDPR) or completed a privacy impact assessment?
- **Iterative Approach**: If you discover new issues mid-way, have you circled back to revise earlier design or risk assessments?
- **Long-Term Sustainability**: As your system grows, will your privacy measures scale effectively (e.g., encryption overhead, user data controls, governance)?

Establishing a culture of iterative privacy review ensures user trust and safeguards your IoT solution over time.

4.11 Common Pitfalls and Key Insights

- **Skipping User Research**: Without real user input, you risk collecting extraneous data or missing critical privacy concerns.
- **Overloading Features**: Resist the urge to add every possible function; more sensors/data often complicates privacy and compliance.
- **Weak or No Consent Mechanisms**: Failing to inform users or secure their explicit permission undermines trust.
- **Ignoring Edge Cases**: Consider offline functionality, intermittent connectivity, or malicious actors physically tampering with devices.
- **Insufficient Iteration**: Privacy design is not a one-time fix; revisiting as your product evolves is essential.

4.12 Conclusion

Workshop 4 has equipped you with the **principles and frameworks of Privacy by Design**, building upon the user-centric foundations of Workshops 1, 2, and 3. Through careful risk mapping, strategic use of a **Privacy Deck**, and iterative refinement, you've embedded privacy considerations into your IoT concept. By reducing data collection, ensuring transparency, and maintaining strong compliance measures, you foster **user trust** and **regulatory confidence** in your system.

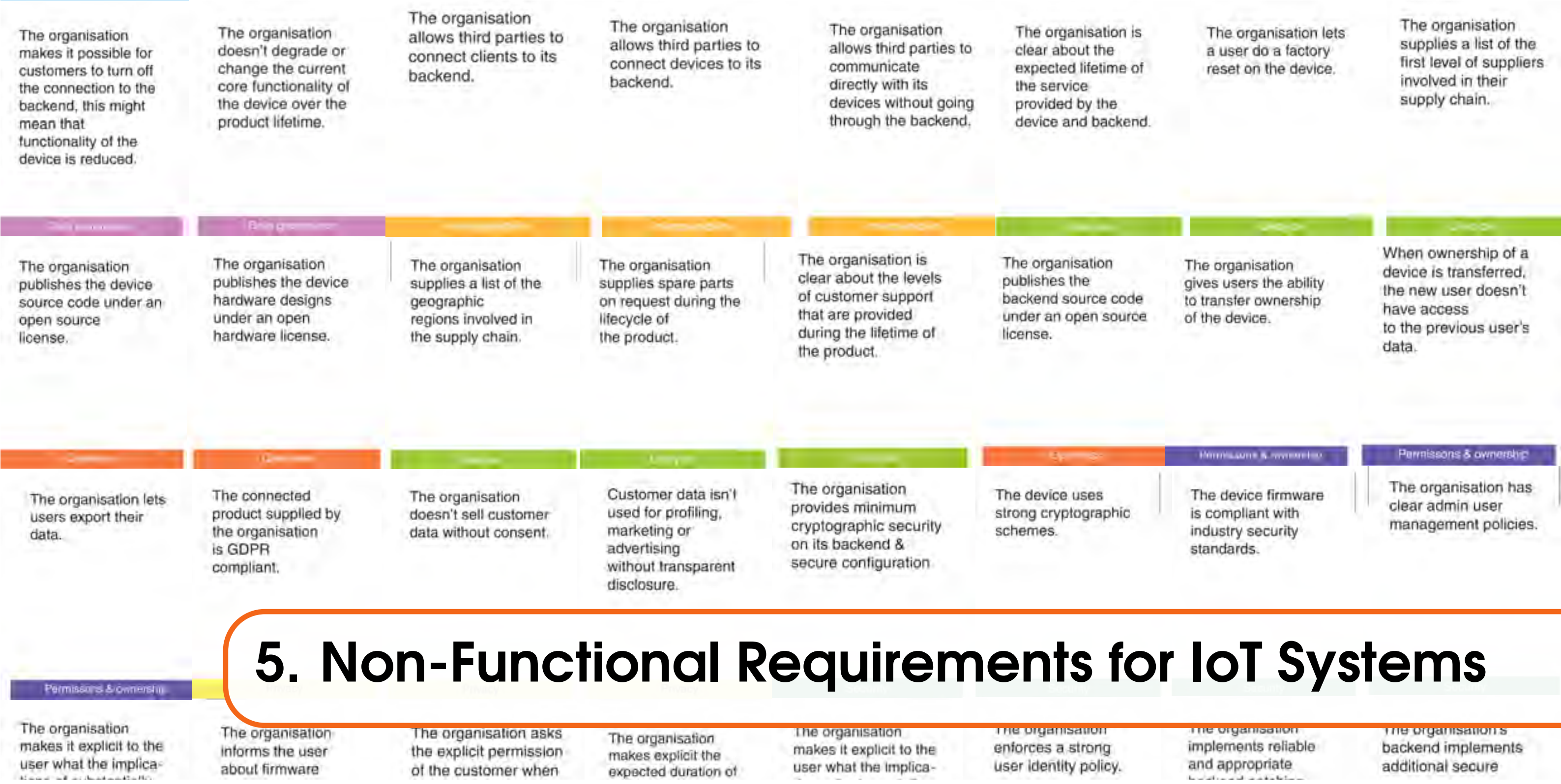
Looking ahead, these privacy-first strategies will dovetail naturally with non-functional requirements or security-hardening steps you may explore in future workshops. By continuously revisiting and testing your design under real-world conditions—be it in a home, city, or industrial environment—you ensure that privacy remains an integral, evolving aspect of your IoT solution.

Table 4.2: List of Privacy frameworks, patterns, and principles

Strategies by Hoepman (2014)		
01. Minimise 04. Aggregate 07. Enforce	02. Hide 05. Inform 08. Demonstrate	03. Separate 06. Control
Strategies by Rost and Bock (2011)		
01. Availability 04. Transparency	02. Integrity 05. Unlinkability	03. Confidentiality 06. Ability to intervene

Principles by Cate (2006)		
01. Notice / Awareness	02. Choice / Consent Choice	03. Access / Participation
04. Integrity / Security	05. Enforcement / Redress	
Principles by ISO/IEC 29100 (2011)		
01. Consent and choice	02. Purpose legitimacy and specification	03. Collection limitation
04. Data minimization	05. Use, retention and disclosure limitation	06. Accuracy and quality
07. Openness, transparency and notice	08. Individual participation and access	09. Accountability
10. Information security	11. Privacy compliance	
Principles by Ann Cavoukian (2009)		
01. Proactive not Reactive; Preventative not Remedial	02. Privacy as the Default Setting	03. Privacy Embedded into Design
04. Full Functionality - Positive-Sum, not Zero-Sum	05. End-to-End Security - Full Lifecycle Protection	06. Visibility and Transparency - Keep it Open
07. Respect for User Privacy - Keep it User-Centric		
Principles by Wright and Raab (2014)		
01. Right to dignity	02. Right to be let alone	03. Right to anonymity
04. Right to autonomy	05. Right to individuality and uniqueness of identity	06. Right to assemble or associate with others without being surveilled
07. Right to confidentiality and secrecy of communications	08. Right to travel (physical/cyber) without tracking	09. No requirement to pay to exercise privacy rights
Principles by Fisk et al. (2015)		
01. Principle of Least Disclosure	02. Principle of Qualitative Evaluation	03. Principle of Forward Progress
Principles by Cavoukian and Jonas (2012)		
01. Full Attribution	02. Data Tethering	03. Analytics on Anonymized Data
04. Tamper-Resistant Audit Logs	05. False Negative Favoring Methods	06. Self-Correcting False Positives
07. Information Transfer Accounting		
Guidelines by O'Leary (1995)		
01. Collection limitation	02. Data quality	03. Purpose specification
04. Use limitation	05. Security safeguards	06. Openness
07. Individual participation	08. Accountability	
Guidelines by Perera et al. (2017)		
01. Minimise data acquisition	02. Minimise number of data sources	03. Minimise raw data intake
04. Minimise knowledge discovery	05. Minimise data storage	06. Minimise data retention period
07. Hidden data routing	08. Data anonymisation	09. Encrypted data communication
10. Encrypted data processing	11. Encrypted data storage	12. Reduce data granularity
13. Query answering	14. Distributed data processing	15. Distributed data storage
16. Knowledge discovery based aggregation	17. Geography based aggregation	18. Chain aggregation
19. Time-period based aggregation	20. Category based aggregation	21. Information Disclosure
22. Control	23. Logging	24. Auditing
25. Open Source	26. Data Flow	27. Certification

28. Standardisation	29. Compliance	
Privacy Patterns		
1. Protection against Tracking	2. Location Granularity	3. Minimal Information Asymmetry
4. Informed Secure Passwords	5. Awareness Feed	6. Encryption with user-managed keys
7. Federated Privacy Impact Assessment	8. Use of dummies	9. Who's Listening
10. Privacy Policy Display	11. Layered Policy Design	12. Discouraging blanket strategies
13. Reciprocity	14. Asynchronous notice	15. Abridged Terms and Conditions
16. Policy Matching Display	17. Incentivized Participation	18. Outsourcing [with consent]
19. Ambient Notice	20. Dynamic Privacy Policy Display	21. Privacy Labels
22. Data Breach Notification Pattern	23. Pseudonymous Messaging	24. Onion Routing
25. Strip Invisible Metadata	26. Pseudonymous Identity	27. Personal Data Store
28. Trust Evaluation of Services Sides	29. Aggregation Gateway	30. Privacy icons
31. Privacy-Aware Network Client	32. Sign an Agreement to Solve Lack of Trust	33. Single Point of Contact
34. Informed Implicit Consent	35. Enable/Disable Functions	36. Privacy Colour Coding
37. Appropriate Privacy Icons	38. User data confinement pattern	39. Icons for Privacy Policies
40. Obtaining Explicit Consent	41. Privacy Mirrors	42. Appropriate Privacy Feedback
43. Impactful Information and Feedback	44. Decoupling [content] and location visibility	45. Platform for Privacy Preferences
46. Access control	47. Pay Back	48. Privacy dashboard
49. Preventing mistakes or reducing their impact	50. Obligation Management	51. Informed Credential Selection
52. Anonymous Reputation-based Blacklisting	53. Negotiation of Privacy Policy	54. Reasonable Level of Control
55. Masquerade	56. Buddy List	57. Privacy Awareness Panel
58. Lawful Consent	59. Privacy Aware Wording	60. Sticky Policies
61. Personal Data Table	62. Informed Consent for Web-based Transactions	63. Added-noise measurement obfuscation
64. Increasing awareness of information aggregation	65. Attribute Based Credentials	66. Trustworthy Privacy Plug-in
67. Selective Disclosure	68. Private link	69. Anonymity Set
70. Active broadcast of presence	71. Unusual Activities	72. Identity Federation Do Not Track Pattern
73. Dynamic Location Granularity		


The organisation makes it possible for customers to turn off the connection to the backend, this might mean that functionality of the device is reduced.
The organisation doesn't degrade or change the current core functionality of the device over the product lifetime.
The organisation allows third parties to connect clients to its backend.
The organisation allows third parties to connect devices to its backend.
The organisation allows third parties to communicate directly with its devices without going through the backend.
The organisation is clear about the expected lifetime of the service provided by the device and backend.
The organisation lets a user do a factory reset on the device.
The organisation supplies a list of the first level of suppliers involved in their supply chain.
The organisation publishes the device source code under an open source license.
The organisation publishes the device hardware designs under an open hardware license.
The organisation supplies a list of the geographic regions involved in the supply chain.
The organisation supplies spare parts on request during the lifecycle of the product.
The organisation is clear about the levels of customer support that are provided during the lifetime of the product.
The organisation publishes the backend source code under an open source license.
The organisation gives users the ability to transfer ownership of the device.
When ownership of a device is transferred, the new user doesn't have access to the previous user's data.
The organisation lets users export their data.
The connected product supplied by the organisation is GDPR compliant.
The organisation doesn't sell customer data without consent.
Customer data isn't used for profiling, marketing or advertising without transparent disclosure.
The organisation provides minimum cryptographic security on its backend & secure configuration
The device uses strong cryptographic schemes.
The device firmware is compliant with industry security standards.
Permissions & ownership
Permissions & ownership
The organisation has clear admin user management policies.
Permissions & ownership
The organisation makes it explicit to the user what the implica-
The organisation informs the user about firmware
The organisation asks the explicit permission of the customer when
The organisation makes explicit the expected duration of
The organisation makes it explicit to the user what the implica-
The organisation enforces a strong user identity policy.
The organisation implements reliable and appropriate
The organisation's backend implements additional secure

5. Non-Functional Requirements for IoT Systems

5.1 Introduction and Preparation

Welcome to **Workshop 5: Non-Functional Requirements for Internet of Things (IoT)**. In this session, you will delve into the *essential* aspects of IoT system performance that extend *beyond* core functionality. By examining **scalability**, **reliability**, **security**, and other non-functional requirements (NFRs), you will learn to build robust, efficient systems that address real-world constraints. Like previous workshops, you can either build on an existing concept from Workshops 1–4 or start fresh with a new idea. Either way, keep in mind that non-functional requirements often **overlap with earlier design and privacy decisions**, so you may need to circle back to earlier assumptions if new insights emerge.

Workshop Goals

By the end of this session, you should be able to:

- **Identify Non-Functional Requirements**: Recognize key NFRs (e.g., security, scalability, availability) that shape IoT solutions.
- **Navigate Trade-Offs**: Balance functional and non-functional needs without compromising core system goals.
- **Refine and Iterate**: Incorporate NFRs into existing IoT concepts, ensuring a holistic, future-proof approach.
- **Enhance System Performance**: Understand how operational constraints and real-world context (budget, environment, regulations) influence IoT deployments.

Recommended Team Roles: We suggest designating a *facilitator* (to keep discussions on track), a *note-taker* (to document how each NFR is handled), and a *timekeeper* (to maintain pacing). Role-playing (e.g., "security officer," "cost-conscious manager," "user advocate") can also expose conflicting priorities early.

Pre-Workshop Checklist

To fully engage in this workshop, you should:

- Have a basic IoT concept or design (from prior workshops or a personal project) that you

plan to refine.
- **Recall any privacy measures** from Workshop 4 or user feedback from Workshops 1–3, as these may shape certain NFRs (e.g., reliability, security).
- Bring sketches, diagrams, or storyboards detailing your IoT system's functionality and known constraints (budget, environment, regulations).

5.2 Objectives

- **Explore Core Building Blocks** of IoT and link them to non-functional requirements.
- **Address Key NFRs** (operation, revision, transition) to ensure a stable, secure, and maintainable system.
- **Learn Collaborative Design Methods** through team activities, rapid prototyping, and iterative feedback loops.
- **Map NFRs to Components** (sensors, gateways, cloud, user interface) to understand their practical impact.

5.3 Learning Outcomes

After completing this workshop, you should be able to:

- **Define and Categorize NFRs** relevant to IoT contexts (e.g., privacy, safety, reliability).
- **Integrate NFR Considerations** into data flow diagrams, storyboards, and prototypes.
- **Resolve Conflicting Requirements** (e.g., security vs. usability) using structured methods and **documented rationale**.
- **Test and Measure NFRs** via simulations or load-testing to validate your design choices.
- **Present a Polished Concept** through a short pitch that shows how NFRs improve performance, user trust, and compliance.

5.4 Related Lessons

Although not mandatory, familiarity with the following lessons (📺) will enrich your understanding:

- **Lesson 2 – Architectures**: Examines how different IoT architectures (centralized, edge, hybrid) affect NFRs.
- **Lesson 5 – Data Management and Analytics**: Shows how data handling, storage, and processing influence performance and scalability.
- **Lesson 6 – Privacy and Security**: Explains confidentiality, integrity, and compliance for robust IoT design.

If you have not completed these lessons, you can still engage with the workshop but may find some topics more challenging.

5.5 Workshop Material Access

Resources **Choose your resources:**

- **Digital Materials**: Access the Miro Board from .
- **Physical Kits**: Download and print the card sets from .

Credits to the original creators: betteriot.org

5.6 Recommended Outcome of the Workshop

By the end of this session, each team should ideally produce:

- A **Revised IoT Design** (diagram or storyboard) that explicitly addresses non-functional requirements.
- A **Data Flow Diagram** highlighting where NFRs like security, reliability, or scalability are most critical.
- A **Short Elevator Pitch**, showcasing how functional and non-functional requirements complement each other for a robust solution.

Measuring Success

Success Measures: As you reach the end of this workshop, consider:

- Have you *identified and documented* **key NFRs** (e.g., security, reliability) and linked them to specific system components?
- Did you **resolve any major conflicts or trade-offs** (e.g., cost vs. performance) by discussing alternatives and noting a final decision?
- Have you **tested or simulated** at least one scenario (e.g., load stress, partial network outage) to confirm NFR feasibility?
- Are your **long-term maintenance** and **scalability** plans clear (e.g., phased upgrades, modular design)?
- *Have you solicited feedback* from peers or end-users to see if NFRs align with real operational needs?

■

5.7 Quick-Reference Table

Table 5.1 outlines the main steps for **Workshop 5**, along with recommended durations and expected outputs. Remember that NFRs often surface new issues, so you may need to revisit earlier steps if priorities or constraints shift.

Table 5.1: Workshop 5 Steps, Activities, Time, and Outputs (Detailed)

Step	Activity	Time	Main Output
1	Identify the Problem and Scope	5–10 min	Concise problem statement + constraints
2	Elicit NFRs Using Principles and Cards	15–20 min	Prioritized list of non-functional requirements (must-have vs. nice-to-have)
3	Map NFRs to System Components	10–15 min	Architecture or data-flow diagram w/ NFR annotations
4	Storyboard the User Experience	10–15 min	User-centric narrative, including normal ops + edge cases
5	Prototype and Solicit Feedback	15–20 min	Refined prototype integrating NFR feedback
6	Handle Conflicts and Trade-Offs	10–15 min	Documented rationale for balancing conflicting NFRs
7	Elevator Pitch Presentation	5–10 min	Short pitch emphasizing how NFRs add value
8	Document, Reflect, and Test Further	5–10 min	Roadmap for ongoing refinement + next steps

5.8 Workshop Overview

In this workshop, you will:

- Brainstorm how non-functional requirements shape your IoT concept.
- Explore operational, revision, and transition NFR categories (see Figure 5.1).

- Create data flow diagrams and storyboards that emphasize NFR considerations.
- Use hands-on activities (interviews, checklists) to uncover hidden or overlooked requirements, including real **market or cost constraints**.
- Present an elevator pitch focusing on how NFRs bolster system performance, user trust, and compliance.

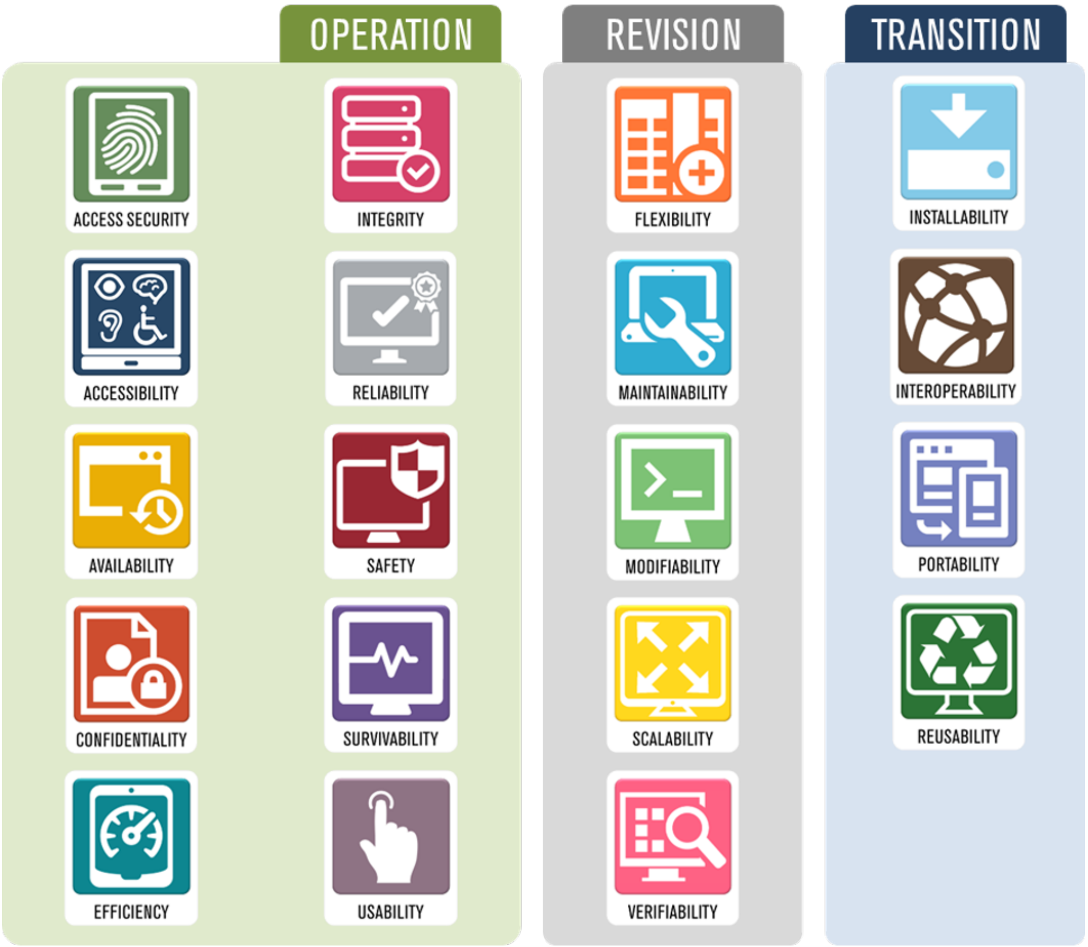


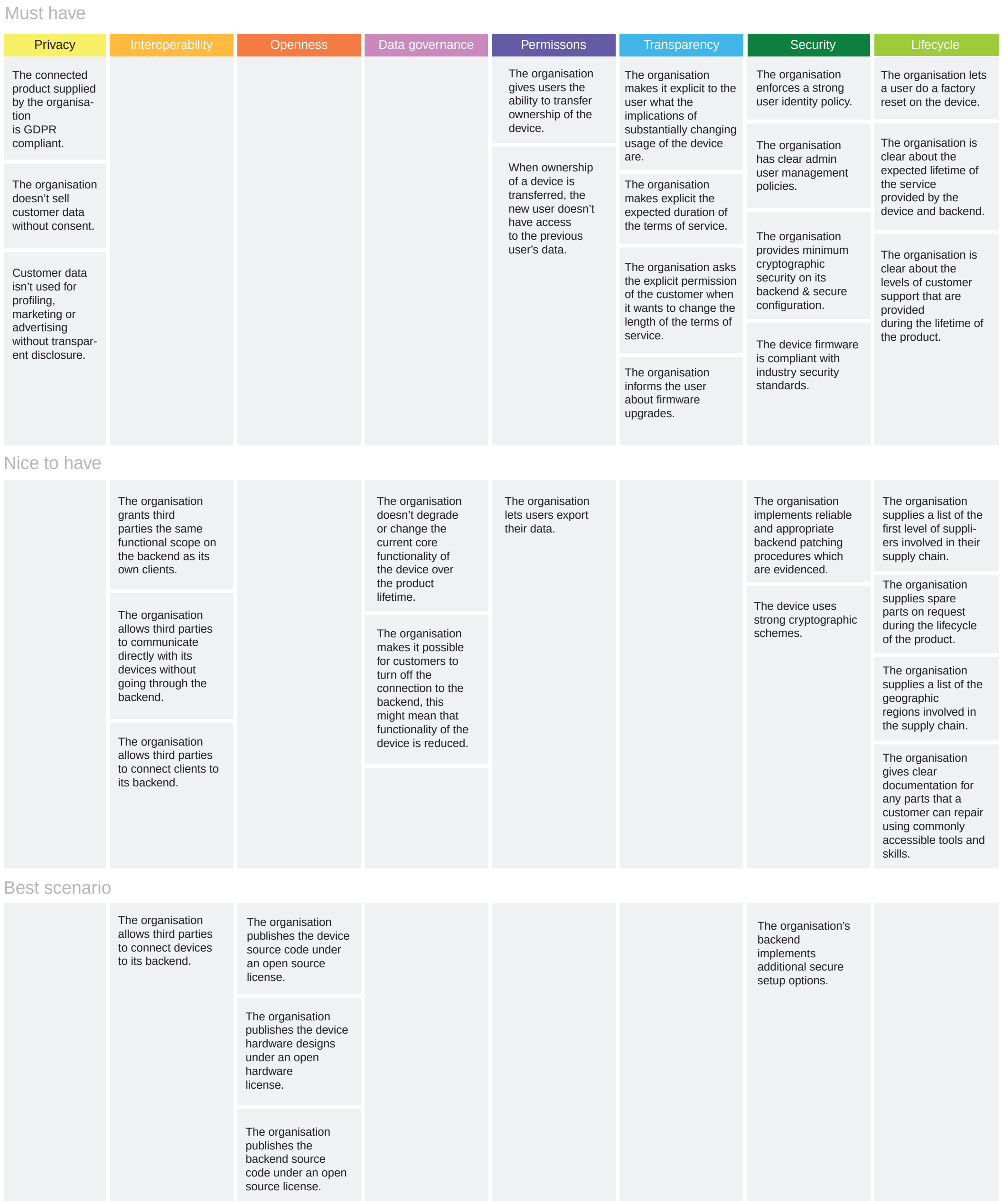
Figure 5.1: This diagram highlights three core categories of non-functional requirements—**Operation**, **Revision**, and **Transition**—each encompassing sub-dimensions like *access security*, *maintainability*, and *interoperability*. **Operational** requirements (e.g., reliability, safety) shape daily system performance, while **Revision** requirements (e.g., maintainability, scalability) support its long-term evolution. Lastly, **Transition** requirements (e.g., portability, interoperability) enable your solution to adapt seamlessly to new contexts, hardware, and stakeholders.

5.9 Step-by-Step Guide

Below is a roadmap for integrating non-functional requirements into your IoT design. You can adapt these steps and timings to your group's experience level. If you uncover new user constraints or discover that certain NFRs conflict, feel free to cycle back to earlier phases.

Step 1: Identify the Problem and Scope (5–10 Minutes)

Clarify Your Use Case: Revisit your IoT concept—whether it's smart home automation, environmental monitoring, or a wearable health device—and outline its core functional requirements (e.g.,



	Privacy	Interoperability	Openness	Data governance	Permissons	Transparency	Security	Lifecycle
Must have	The connected product supplied by the organisation is GDPR compliant. The organisation doesn't sell customer data without consent. Customer data isn't used for profiling, marketing or advertising without transparent disclosure.				The organisation gives users the ability to transfer ownership of the device. When ownership of a device is transferred, the new user doesn't have access to the previous user's data.	The organisation makes it explicit to the user what the implications of substantially changing usage of the device are. The organisation makes explicit the expected duration of the terms of service. The organisation asks the explicit permission of the customer when it wants to change the length of the terms of service. The organisation informs the user about firmware upgrades.	The organisation enforces a strong user identity policy. The organisation has clear admin user management policies. The organisation provides minimum cryptographic security on its backend & secure configuration. The device firmware is compliant with industry security standards.	The organisation lets a user do a factory reset on the device. The organisation is clear about the expected lifetime of the service provided by the device and backend. The organisation is clear about the levels of customer support that are provided during the lifetime of the product.
Nice to have		The organisation grants third parties the same functional scope on the backend as its own clients. The organisation allows third parties to communicate directly with its devices without going through the backend. The organisation allows third parties to connect clients to its backend.		The organisation doesn't degrade or change the current core functionality of the device over the product lifetime. The organisation makes it possible for customers to turn off the connection to the backend, this might mean that functionality of the device is reduced.	The organisation lets users export their data.		The organisation implements reliable and appropriate backend patching procedures which are evidenced. The device uses strong cryptographic schemes.	The organisation supplies a list of the first level of suppliers involved in their supply chain. The organisation supplies spare parts on request during the lifecycle of the product. The organisation supplies a list of the geographic regions involved in the supply chain. The organisation gives clear documentation for any parts that a customer can repair using commonly accessible tools and skills.
Best scenario		The organisation allows third parties to connect devices to its backend.	The organisation publishes the device source code under an open source license. The organisation publishes the device hardware designs under an open hardware license. The organisation publishes the backend source code under an open source license.				The organisation's backend implements additional secure setup options.	

Figure 5.2: This outcome table illustrates three distinct levels of non-functional requirements—**must-have**, **nice-to-have**, and **best scenario**—spanning considerations like *privacy*, *security*, and *data governance*. Each row shows how organizations can incrementally enhance their IoT solutions by integrating more advanced features or openness. As depicted in Figure 5.2, these categories are mapped to varying degrees of readiness. For instance, enforcing strong user identity policies addresses the "must-have" level of security, whereas adopting open-source hardware designs and sophisticated cryptographic options exemplifies the "best scenario."

measuring temperature, sending alerts).

- **Stakeholder Interviews**: Talk to potential users or teammates to uncover assumptions about reliability, security, or data ownership.
- **Constraints**: Note budgets, schedules, or technical limits that might affect feasibility of certain NFRs (e.g., if hardware is low-power).

Outcome: A *concise problem statement* clarifying your top-level objectives and constraints—laying the foundation for your NFR considerations.

Step 2: Elicit NFRs Using Principles and Cards (15–20 Minutes)

Refer to **Open IoT Principles** or an **NFR card set** to systematically address privacy, interoperability, data governance, performance, and other key areas (Figures 5.1 and 5.2). Recall any privacy constraints from Workshop 4 or user feedback from prior modules.

- **Operation NFRs**: Security, reliability, availability, accessibility.
- **Revision NFRs**: Maintainability, modifiability, scalability.
- **Transition NFRs**: Installability, portability, interoperability.

Group Brainstorm:

- Discuss which NFRs are *must-have* vs. *nice-to-have*.
- Consider "best scenario" options if cost or user acceptance are not limiting.
- Note potential conflicts (e.g., "extreme reliability" vs. "limited budget").

Outcome: A *prioritized list of NFRs* guiding design and technical decisions.

Step 3: Map NFRs to System Components (10–15 Minutes)

Create or refine a **data flow diagram** or **system architecture chart** showing how information travels between components (sensors, gateways, cloud, user interface). Indicate where and how each NFR applies:

- **Security Zones**: Mark sections needing encryption (e.g., device-to-cloud) or authentication (e.g., user login).
- **Scalability Points**: Identify places (databases, message brokers) that might become bottlenecks under high load or real-time demands.
- **Maintainability Hooks**: Show how modules can be updated without bringing down the entire system. This helps plan for future expansions.

Outcome: A *detailed architecture diagram* with annotations on where NFRs (Figures 5.1 and 5.2) are crucial.

Step 4: Storyboard the User Experience (10–15 Minutes)

Develop a **storyboard** that focuses on user interactions under normal and abnormal conditions (e.g., network dropouts, sensor failures). Weave in NFR references from your priority list:

- **Operational Context**: Does reliability matter more during peak usage? Are there safety-critical interactions (e.g., in healthcare)?
- **Feedback Loops**: If a device malfunctions, how does the user get alerted? Is there a fallback or local mode?
- **User Perception**: Are error messages clear enough to maintain trust? Do load times or latencies degrade the user experience?

Outcome: A *user-centric narrative* illustrating how NFRs shape daily operations and edge cases.

Step 5: Prototype and Solicit Feedback (15–20 Minutes)

Translate your diagrams and storyboards into a **simple prototype** (paper sketches, wireframes, or minimal hardware). Invite peers or stakeholders to provide feedback on key NFR aspects:

- **Performance Tests**: Try simulating multiple devices generating data. Are response times acceptable under typical load?
- **Security Review**: Do you have encryption at rest, strong identity policies, or intrusion detection for critical points?
- **Usability Trials**: Are there friction points where advanced security measures hamper user experience?
- **Real-World Constraints**: If the device is battery-powered, does your reliability approach conflict with power usage?

Outcome: A *refined prototype* addressing top-priority NFRs and validated through quick user or stakeholder tests.

Step 6: Handle Conflicts and Trade-Offs (10–15 Minutes)

Use a decision table or weighting matrix to resolve tensions (e.g., high security vs. minimal onboarding steps). Emphasize iterative reconciliation—these trade-offs may resurface later. Document each compromise and rationale:

- **Security vs. Usability**: Possibly add multi-factor authentication for critical features but allow simpler login for low-risk tasks.
- **Cost vs. Scalability**: Decide if you need high-throughput servers now or can plan a phased upgrade approach.
- **Maintenance vs. Performance**: More modular code can slow initial performance but greatly ease future updates.

Outcome: A *clear rationale* explaining how your team balanced conflicting NFRs, acknowledging future reevaluation if conditions change.

Step 7: Elevator Pitch Presentation (5–10 Minutes)

Craft a short (3-minute) pitch focusing on how non-functional requirements elevate your solution beyond basic functionality:

- **Problem Statement**: Recap the user need or pain point.
- **NFR Highlights**: Emphasize operational, revision, and transition categories tackled.
- **User Benefits**: Show how reliability, security, or performance lead to trust and seamless experiences.
- **Market or Cost Dimension**: If relevant, highlight how you managed or planned for budget constraints or monetization without compromising essential NFRs.

Outcome: A brief, *compelling pitch* underscoring why your system's NFR-conscious design adds tangible value.

Step 8: Document, Reflect, and Test Further (5–10 Minutes)

Capture final outputs (photos, screenshots) of your diagrams, prototypes, and notes. Then:

- **Write a Short Reflection**: Summarize how NFR priorities evolved your design. Mention any persistent open questions.
- **Plan Next Steps**: Outline more advanced testing (stress tests, real hardware trials), potential *ethical or regulatory* constraints, or cost modeling.
- **Iteration Path**: Schedule a future session or checkpoint to revisit NFRs as user demands or tech constraints shift.

Outcome: A *documented roadmap* for ongoing refinement. Real-world IoT systems often require continuous re-checking of NFRs under new conditions (e.g., scaling user base).

5.10 Additional Tips and Next Steps

- **Measurement and Validation**: Use load-testing tools or mock datasets to see if your design meets performance or reliability benchmarks. Keep logs to analyze if NFR targets are consistently hit.
- **Regulatory and Ethical Concerns**: Consider GDPR, FCC, or other frameworks, especially around data sovereignty or radio spectrum usage.
- **Post-Workshop Follow-Up**: Revisit your prototype, incorporate stakeholder feedback, and re-check your NFR list whenever new features are added or user needs change.
- **Role-Playing Exercises**: Let each team member represent a different viewpoint (security, cost, user-friendliness, privacy) to unearth hidden conflicts early.

- **Documentation Templates**: Maintain an NFR matrix linking each requirement to system modules or user stories. Update it as your system grows.
- **Real-World Inspirations**: Look into large IoT projects (e.g., city-wide sensor networks or industrial automation) to see how they handle reliability, scale, and cost constraints in practice.

5.11 Common Pitfalls and Key Insights

- **Insufficient Iteration**: NFRs are not a one-time check; discover new conflicts or demands and circle back to refine.
- **Ignoring Real-World Constraints**: Overlooking budgets, environment, or regulatory conditions can render a design infeasible.
- **Neglecting User Perspective**: Some NFR solutions may degrade usability if not balanced with user feedback.
- **Minimal Viable vs. Feature Overload**: Start with essential NFR goals; more advanced features can be phased in later once resources or user acceptance grow.
- **Forgetting Long-Term Monitoring**: NFR success can shift over time—monitoring or analytics can reveal if reliability or performance degrade under new loads.

?

5.12 Conclusion

Non-functional requirements play a **pivotal** role in shaping IoT solutions that are *secure, scalable, and user-friendly*. By proactively addressing these constraints—through data flow analysis, storyboarding, and iterative prototyping—you ensure your system can handle real-world demands. As Figure 5.1 demonstrates, operational, revision, and transition NFRs each target different life-cycle dimensions, while Figure 5.2 shows how organizations can evolve from “must-have” to “best scenario” features.

Embracing this holistic approach empowers you to balance functional goals with deeper considerations like performance, reliability, privacy, and adaptability. Whether you decide to expand your concept with advanced security measures, new data analytics, or integrated privacy controls, these non-functional requirements will remain a guiding force as you refine, test, and scale your IoT solution over time.