Design for IoT Devices

Designing for Data

Importance of Data in IoT Design

- IoT devices exist to **generate, process, or respond to data**.
- Data choices influence sensors, storage, processing location, and user/system interactions.

Data Flow in IoT Systems

- **Device → Bridge → Network (Cloud/Fog)**.
- **Cloud**: centralized processing/storage.
- **Fog/Edge**: local/near-device processing before forwarding.
- *Example*: Robot vacuum → on-device navigation + cloud diagnostics.

Types of Data

- **Categorical**: nominal (labels), ordinal (ranked).
- **Numerical**: discrete/continuous; interval vs. ratio scales.

Sources of Data in IoT

- **Static** (rarely changes; context) vs. **Dynamic** (sensor streams; needs filtering/aggregation).

Data Origins: Declared, Direct, Inferred

- **Declared** (user-provided; privacy-sensitive).
- **Direct** (sensor-measured).
- **Inferred** (derived patterns; e.g., occupancy from power data)

Scale of Data: Small vs. Big

- **Small**: single device → local actions.
- **Big**: distributed devices → aggregated insights.

Timeframe of Data Use

- **Real-time** actions vs. **Historical** trend/anomaly analysis.

Context vs. Actions in Data

- **Context** (environment/user/device state) vs. **Action** (alarms, automations).
- *Example*: Smoke detector monitors (context) → alarm (action).

Feedback Loops in IoT

- Observe → Detect → Present → React → Repeat.
- Emphasize critical signals, timeliness, decision latency.

Progressive Disclosure

- Layer info: **critical → supporting → logs/history** to reduce overload.

Data as a Core Element

- Decide where to process (device/edge/cloud), how to transmit/store/secure, and what outcomes the data must drive.

Broader Design Perspectives

Emotional Design (Don Norman)

- **Visceral** (first impression/appearance), **Behavioral** (usability/experience), **Reflective** (identity/status).
- *Example*: Juicy Salif—visceral/reflective strength can outweigh behavioral quirks.

Interaction & Placement

- **Inconspicuous + rare** (closet NAS): maximize clarity of rare controls.
- **Conspicuous + rare** (motion sensor): aesthetics/brand language.
- **Conspicuous + frequent** (TV remote): aesthetics **and** highly legible function.

Visual Brand Language

- Cohesive shapes/materials/colors/graphics → loyalty, recognition, product longevity.
- Style guides/specs enforce consistency (Apple, Dyson).

Phases: Story, Art, Science

- **Story** (form/function/usability), **Art** (aesthetics/sensory feel), **Science** (materials/manufacture/maintenance).

Consistency

- **Visual**, **Functional**, **Internal** (within product line), **External** (with other systems).
- Inconsistency → confusion/frustration; consistency → trust/learnability.

Data Analysis & AI for IoT Networks

IoT Data Characteristics

- **Structured** (DB/spreadsheet) vs. **Unstructured** (~80%: text/audio/video/images).
- **Data in Motion** (streams at edge) vs. **Data at Rest** (stored/batch).

Analytics Roles

- **Descriptive → Diagnostic → Predictive → Prescriptive** (increasing complexity/value).
- Real-time anomaly/trend detection is especially valuable.

Cloud vs. Edge

- **Cloud**: Hadoop/MapReduce, large-scale ML/text mining.
- **Edge/Fog**: reduce bandwidth, respond faster, time-sensitive control.

Why AI/ML in IoT

- Multivariate, high-volume, multi-source data needs modeling for patterns and actions.
- AI traits: **intentionality, intelligence, adaptability**.

ML Methods

- **Supervised**: labeled training → classification/regression (e.g., "person detected?").
- **Unsupervised**: hidden structure/clusters/anomalies (e.g., factory defect patterns).

Applications
- Monitoring, optimization, predictive maintenance, self-healing/self-optimizing systems.

Big Data "3Vs"
- **Volume, Velocity, Variety** (device/machine/social/enterprise sources).

Edge Tools & Prototyping
- **AWS Greengrass**: train in cloud → deploy inference at edge (filter/transform/correlate/pattern-match locally).
- **TensorFlow Lite**: on-device ML (classification/detection/pose) on SBCs like Raspberry Pi.

Takeaway
- Combine cloud-scale analytics with edge ML for bandwidth efficiency, speed, and resilience.

Designing for Wearables

What Counts as a Wearable
- Embedded electronics worn on body: watches/trackers/glasses/cameras/hearing aids.
- **General-purpose** (Apple Watch) vs. **single-purpose** (hearing aids).

Interaction Modes & Core Considerations
- **Glanceable**, **multi-modal** (visual/audio/haptic/touch).
- Avoid **data avalanche**; privacy-aware in public contexts.
- **Graceful degradation** when offline.

Human-Centered Focus
- Solve recurring human problems; augment good behaviors and automate undesirable tasks.

Enablers
- Moore's Law, ubiquitous connectivity (cellular/Wi-Fi/BLE), open-source ecosystems.

Human Factors: Input/Output
- **Input**: small surfaces, gestures, VUI, passive sensing.
- **Output**: active (alerts/haptics) vs. passive (subtle cues).

Applications & Market
- Fitness/health, smart clothing, AR glasses (privacy lessons from early attempts), body cams, mixed reality, medical wearables (connected inhalers).

Methods & Tools
- **Double Diamond** (diverge/converge research & design, test/refine).
- Journey maps, service blueprints, ML-enhanced sensing, rapid prototyping (SBCs).

Trends & Opportunities

- Smartwatches, fitness trackers, mixed reality, e-textiles, skin patches → growing niches; market maturing but still "wild west" with room for impact.

Key Challenges

- Battery life, sensor fusion, comfort/acceptance, privacy, packaging and durability.

Voice User Interfaces (VUIs)

What is a VUI & Where It's Used

- Voice/speech as an input modality parsing **conversational commands** (not rigid IVR trees).
- Common platforms: Siri, Google Assistant, Alexa, Cortana; contexts: home devices, phones, wearables, TV remotes, toys/games, cars.
- Rapid adoption over the last decade; widely used for general queries, music/media, timers/alarms, reminders/calendar, and home automation.

Pros & Cons

- **Advantages**: hands-free, fast, intuitive (no "training" needed), can feel personable, works on tiny devices (no screen/keyboard needed).
- **Limitations**: noise/public settings, social comfort, privacy (spoken aloud), often **single-turn** interactions, and **accessibility gaps** (consider alternatives for users with hearing/speech/motor impairments).

Core Design Anatomy

- **Intent**: the user's goal ("set a timer").
- **Utterance**: the phrasing ("set a 5-minute timer").
- **Slots**: extracted variables (duration=5 minutes), **required vs. optional**.
- **Dialog Flow/Map**: prototype expected paths, branches, confirmations, and error recoveries.
- **Sample Dialogs**: cover target use-cases and edge cases; storyboards/wireflows help visualize turns.

Tools & Prototyping

- Vendor & OSS ecosystems for design/dev; pick based on deployment target.
- **Fast prototype** example: Raspberry Pi + mic/speaker + **Alexa AVS Device SDK**; extend with custom skills (e.g., GPIO control).
- Cloud STT/TTS/NLU stacks available across vendors; choose based on accuracy, latency, cost, languages, and integration.

VUI Heuristics & Principles

- **General UX heuristics apply**: user control & freedom (say "stop/cancel"), error prevention/handling, minimalist phrasing, helpful feedback.
- **VUI-specific guidance**:
    - Consider **multimodal cues** (lights/display ring) for state ("listening…/thinking…").

    - Prefer **one-turn** tasks; for multi-turn, maintain **short-term memory** (handle pronouns, references).
    - Provide clear **acknowledge/cancel** patterns and **opposite tasks** (on/off).
    - Offer **examples** when asking for structured info ("say: 'add milk to my shopping list'").
    - Personalize lightly (names/pronunciation) without forcing a "persona."

Error Taxonomy & Recovery

- **Heard speech present but unrecognized** (noise/ASR fail).
- **Recognized but unsupported** (out-of-scope intent).
- **Misclassification** (wrong intent/slots).
- **Escalation failures** mid-dialog (timeouts, barge-ins).
- Design **graceful fallbacks**: reprompts, confirmations on low confidence, alternative phrasings, and handoff to visual help when available.

Vendor/Stack Selection Criteria

- **ASR/NLU quality** (accuracy, domain vocabulary/"N-best" results).
- **Endpointing** (start/stop-of-speech), **barge-in** handling.
- **Timeout policies** (too-much-speech, silence).
- **Latency** and offline/local options.
- **Language/accent coverage**, personalization, cost, privacy/compliance.

Testing & Measurement

- **Usability tests** on representative tasks; surveys for satisfaction.
- **Quantitative KPIs**: task completion, abandonment/dropouts, average dialog time, errors per session, confirmations, latency.
- **Common issues to catch**: missing/incorrect/repeating prompts, awkward TTS phrasing, data fetch latency, context loss.

Avatars & the Uncanny Valley

- Avatars can humanize VUIs but risk discomfort near human-like realism.
- Prefer **abstracted** visuals (lights, simple animations, icons) over hyper-realistic faces unless you can meet user expectations consistently.

Key Takeaways

- VUIs are now an expected modality; apply classic UX plus VUI-specific practices.
- Favor **clear, short, recoverable** dialogs; support accessibility and privacy.
- Leverage cloud vendor stacks for rapid prototyping; optimize for your domain.
- Data drives IoT (types, flows, feedback/progressive disclosure).
- Design perspectives (emotional design, placement, brand language, product phases, consistency).
- AI/analytics (cloud vs. edge; supervised vs. unsupervised ML).
- Wearables (human factors, interaction modes, market/applications, methods).
- VUIs (anatomy, heuristics, tools, testing, avatars, accessibility).
- **Final Note**: Successful IoT products blend **data strategy, human-centered design, robust system engineering, and appropriate AI/VUI choices**.