

IOT ROADMAP

SUPPORT FOR INTERNET OF THINGS
SOFTWARE SYSTEMS ENGINEERING



PROPOSAL

What to consider when engineering IoT



FOCUS

IoT development teams



AUTHORS

Rebeca Motta, Káthia Oliveira and Guilherme Travassos
(COPPE/UFRJ) (LAMIH/UPHF) (COPPE/UFRJ)

ABOUT THE RESEARCH

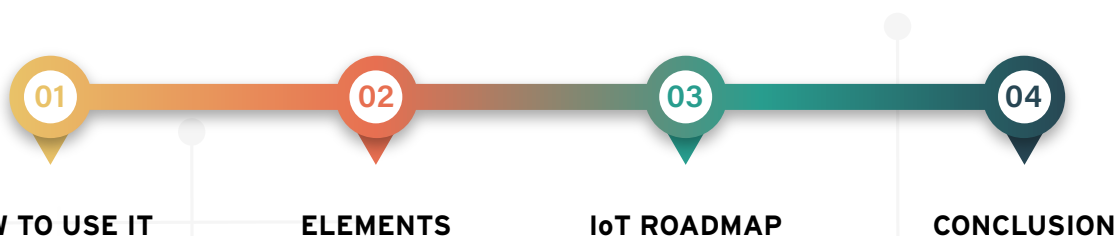
The Roadmap is performed in the context of a Ph.D. research in collaboration between the Experimental Software Engineering Group, from the Systems Engineering and Computing Program of the Federal University of Rio de Janeiro (COPPE/UFRJ) and the Laboratory of Industrial and Human Automation Control, Mechanical engineering and Computer Science (LAMIH UMR CNRS 8201) in the Université Polytechnique Hauts-de-France (UPHF).

To cite this work please refer to: **MOTTA**, Rebeca Campos; **OLIVEIRA**, Káthia Marçal; **TRAVASSOS**, Guilherme Horta. "Towards a Roadmap for the Internet of Things Software Systems Engineering". <https://doi.org/10.1145/3415958.3433100>

ABOUT THE PROPOSAL

The Roadmap resulted from an investigation on the particularities of IoT applications. It is the concrete organization of the concepts and evidence gathered from different experimental studies. It comes to support the definition of IoT software systems, with specific items for the project team to discuss and define the essential aspects related to the specifying, designing, and implementing an IoT application.

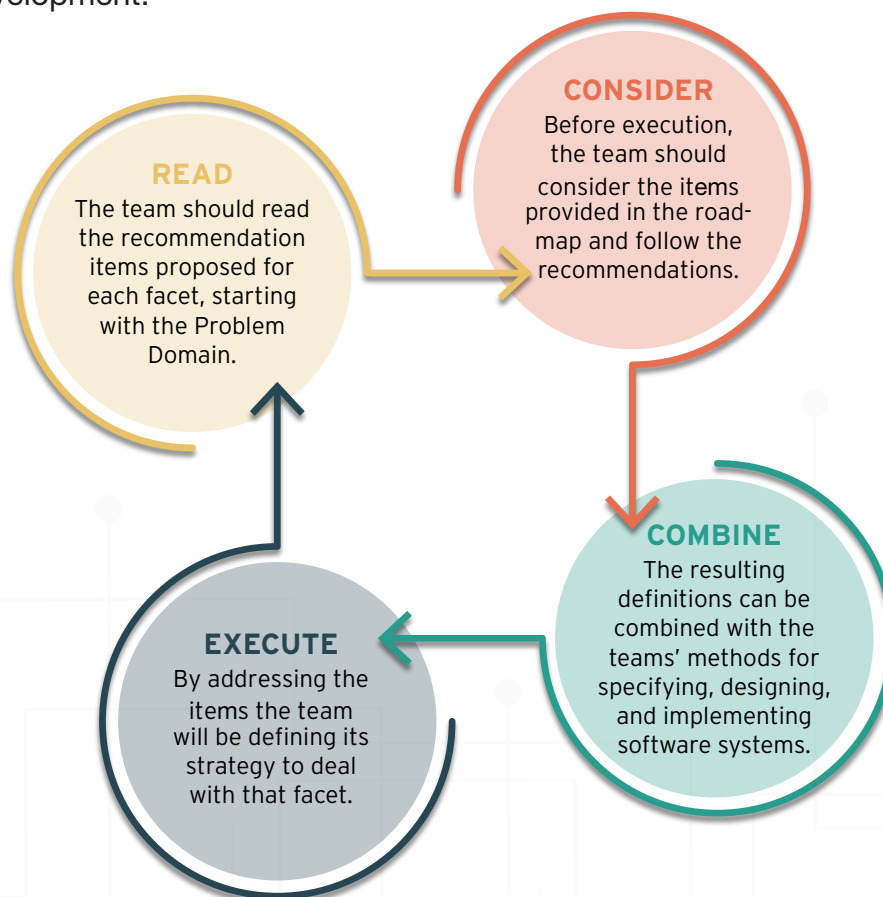
CONTENT



HOW TO USE IT

The team should **(1) read** the Roadmap to encourage discussions of the details related to each facet. They should **(2) consider** each question before following the recommendations. This way, the understanding of the items is aligned among all the team. By the **(3) execution** they establish their strategy for the project. The Roadmap does not aim to replace everyday activities in the development or the original methods. Nevertheless, the roadmap can be **(4) combined** with the existing methods and technologies already in use.

The Roadmap can be used iteratively, following the project cycles. With this, it is possible to minimize the project uncertainty. All stakeholders can use it as a guide to support discussions and decision-making for directions to an action plan for the development.



The items are grouped in categories. Each item can be marked as **Done** - if it is already completed, **To Do** - if it is an activity for the next phases, and **Not Applicable (N/A)** - if it is not in the project plan.

Please pay attention to **Cross-cutting items** since they can evolve and change throughout the project. These items are marked with:



IoT ROADMAP: THE ELEMENTS

The structure of the roadmap combines phases, facets, and items.

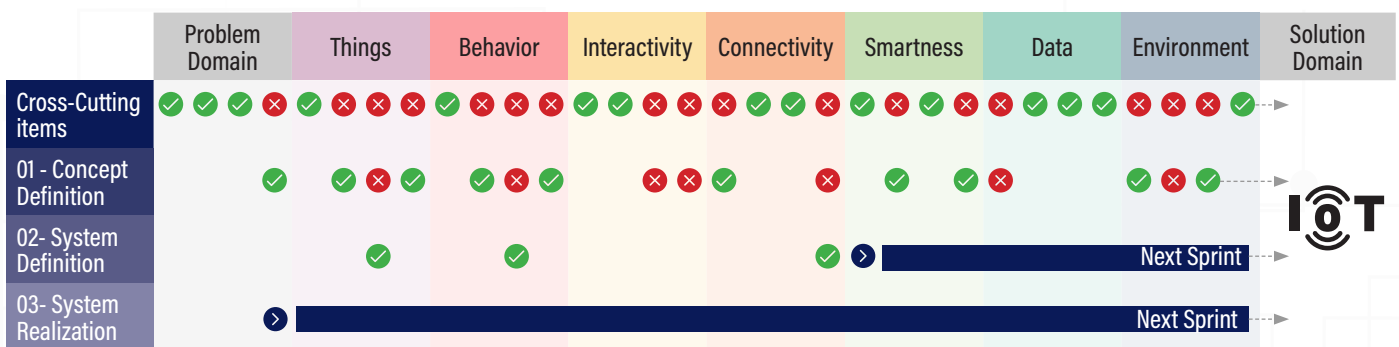
The **phases** organize the engineering life cycle through time. It goes from the need for an IoT product to the construction of the product itself.

In IoT, several areas are intertwined to achieve a solution. Therefore the phases should be considered in a multi-faceted way to address the IoT requirements in a multidisciplinary fashion with the **facets**.

Each facet is composed of a set of **items**, which represent activities, definitions and recommendations for the project team to achieve the desired solution.

PHASES	FACETS	ITEMS
<i>A generic life cycle with three phases</i>	<i>Disciplines and knowledge areas involved in IoT</i>	<i>Recommendations for IoT particularities</i>
From conception to realization, the phases organizes the activities to transform requirements into a deliverable IoT software product.	IoT software-based solutions involve seven facets that require different skills and technologies to go from the problem to the solution domain.	Activities to support the project team to discuss and define aspects related to the specifying, designing, and implementing an IoT application.

By following the Roadmap, as the project progress in the phases, it will be possible to deal with different details and concerns for each facet in an IoT project.



IoT ROADMAP: THE PHASES

Based on the Body of Knowledge to Advance Systems Engineering

Systems engineering presents “an interdisciplinary approach and means to enable the realization of successful systems. Successful systems must satisfy the needs of their customers, users, and other stakeholders”. It concerns activities to discover, create and describe a system to satisfy an identified need. The activities are grouped and described as general processes that cover build artifacts, decisions for concept definition, the needs and requirements of stakeholders, and preliminary operational concepts. It considers a generic life cycle with the following phases:

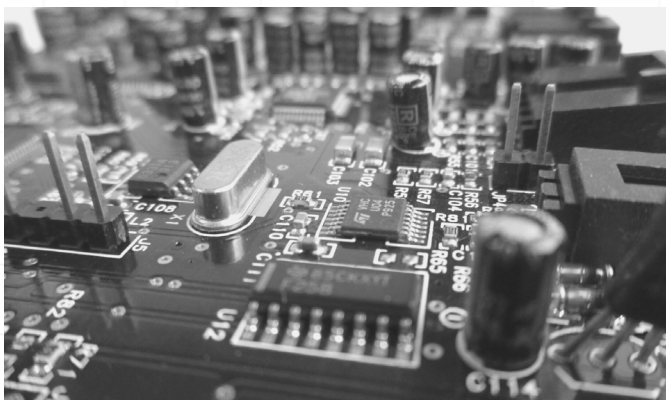


CONCEPT DEFINITION

Where there is a decision to invest resources in a new or improved engineered system. Consists of developing the concept of operations and business, determining the key stakeholders, requirements and the system desired capabilities.

SYSTEM DEFINITION

Evolution and formalization of the requirements to be sufficiently well defined. It provide a basis of system realization considering the architecture and design, compatibility and feasibility of the resulting system definition.



SYSTEM REALIZATION

Aimed to deliver operational capability with the construction as well as the integration of the elements. Verification and validation, preparing for the production, support and utilization activities are included.

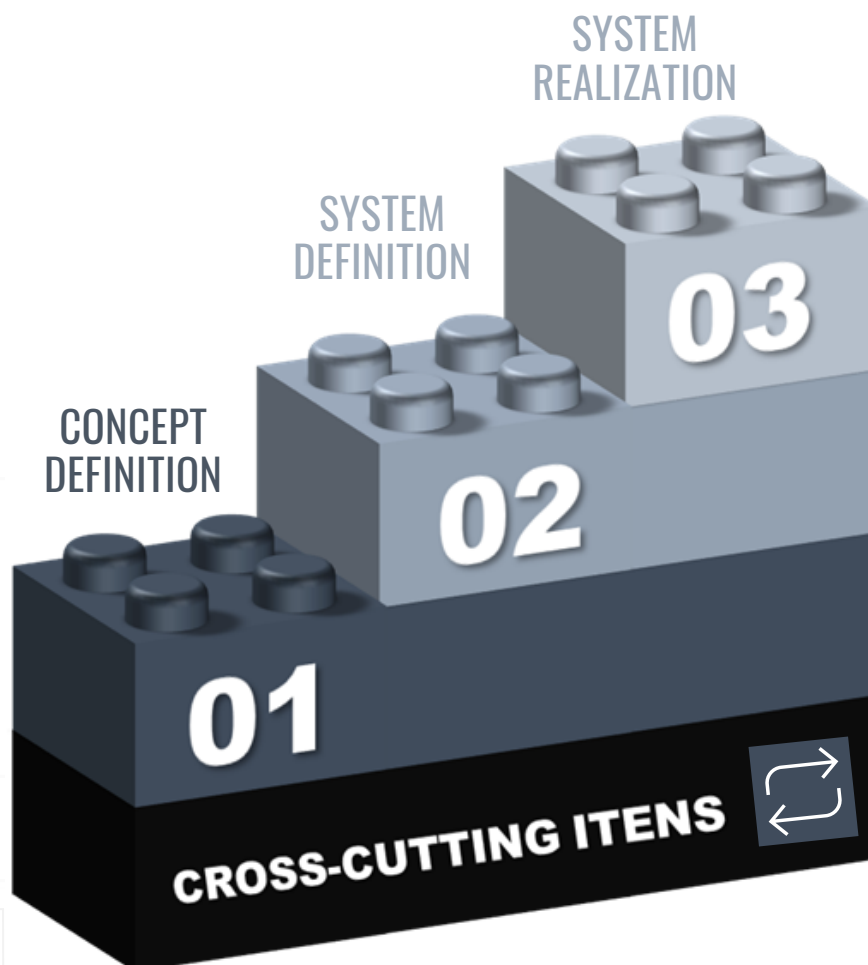
IoT ROADMAP: THE PHASES

Based on the Body of Knowledge to Advance Systems Engineering

The early-stage decisions guide the product being specified, designed, and engineered. They clarify the overall scope and establish a basic understanding of the problem, the people who seek to solve the problem, the type of solution desired, and the team that will oversee the solution.

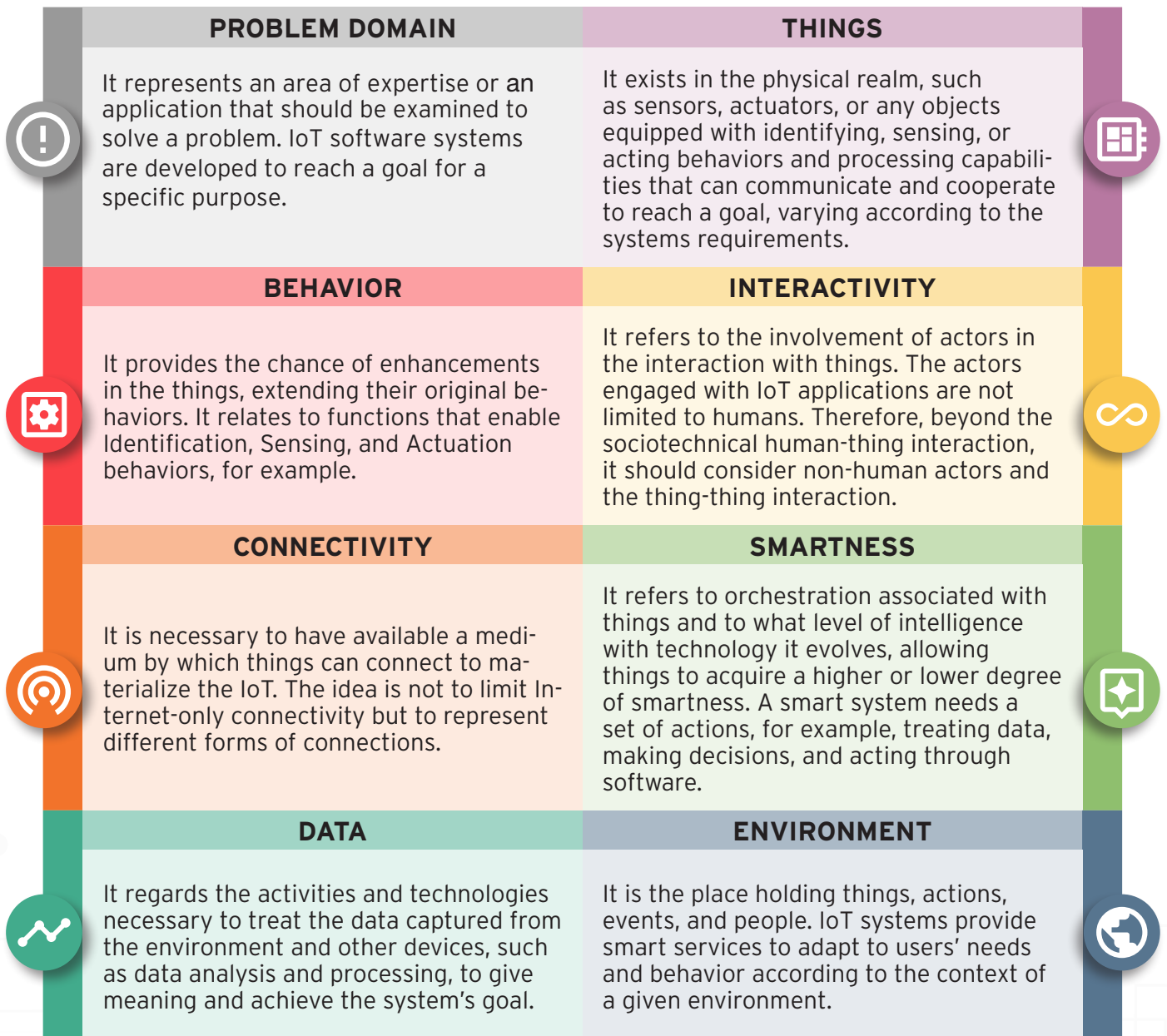
The Roadmap contemplates three phases with specific items and include crosscutting items that are not fixed to a specific phase. These items can be revisited and evolved throughout the project, considering new information that can update the implementation.

The revision and changes of the phases should follow the teams' internal direction for development cycles, such as sprints.



IoT ROADMAP: THE FACETS

The Roadmap aims to address some of the IoT's existing challenges and its multidisciplinary nature. The facets represent different disciplines and knowledge areas involved in IoT. Meaning "one side of something many-sided" (Oxford Dictionary), "one part of a subject, a situation that has many parts" (Cambridge Dictionary), representing the multidisciplinary nature required in such systems.



The problem domain will direct and contextualize how the facets will be derived, implemented, and managed. The Roadmap helps to go from the problem domain to a software solution, considering all the facets as part of the same solution, one related to the other aiming at the completion.



IoT ROADMAP

To be used to support problem understanding and the formal definition for conception and realization towards an IoT solution.



PROBLEM DOMAIN

1. Define the motivation for the IoT project. An IoT based solution is provided for a particular goal, based on a real problem and motivation. The motivation behind the solution could affect how the problem is addressed.



	DONE	TO DO	N/A
Define the problem domain, highlighting the need for IoT solution (such environmental control with real-time actuation).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define the system goal, highlighting the IoT characteristics (such as communication in real-time, wider range and scale, and remote control).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the problem motivation for using IoT technology (such as optimization of resources and requirement for less human intervention).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

2. Define IoT system behavior. Define the basis for the project, defining what the stakeholders - users, things, developers, actors - need from it and what the system must do to satisfy this need. Be well understood and defined by everybody, capture the idea of the product.

	DONE	TO DO	N/A
Define high-level IoT requirements (such as the use of sensors and tags to address sensing).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe high-level IoT behaviors (such as sensing the context of a given environment and actuation in a production line).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify high-level users, roles, actors (such as external service to contribute with information, and users with permissions to adjust the actuation rules).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the high-level context of use (such as a during cropping season in a farm, healthy control in a water tank, and a maintenance in a production site).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



3. Define IoT system limits. A specification or technical limitation to achieve some functionality. It refers to what was defined and what the system doesn't do. This limitation can lead to recommendations for improvements.

	DONE	TO DO	N/A
Establish the IoT technical limitations (such as the size of the solution not to disturb the animals on a farm, and the need to be waterproof in a water tank).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the IoT functional limitations (such as the system will only capture soil information, not weather information, and act based on the human decision, not automatically).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

4. Verify existent IoT solutions. To decide on building or adapting a component, supporting the decision to develop a new system, or being aware of current technologies and available options. Prior research is required to verify existing solutions.

	DONE	TO DO	N/A
Describe existent IoT systems or products (such as similar products to what is going to be developed).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe existent technologies for IoT (such as check if any add-on or component is already available).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

5. Define solution benefits and risks. The proposed IoT solution can achieve the expected goal and deliver some advantages from other alternatives. However, it can also have a downside and possibility of damage, loss, difficulty or threats generated from the IoT solution.

	DONE	TO DO	N/A
Define the benefits of using the IoT solution (such as immediate assistance due to real-time controlling and less human intervention due to smartness).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe and implement mechanisms to mitigate the risks (such as to define regulatory compliance and control access both physical and virtual to the IoT solution).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the possible risks (such as user and product safety).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>





6. Define strategy for relevant quality characteristics and attributes.

The project should have a clear definition of their quality characteristics (assigned property) and attributes (inherent property) in a compliant way with the specification and general expectations. Establish practices to ensure the overall quality and constraints of the system. From the high-level attributes identified as system goals, refine to manageable and measurable items.

Some quality characteristics examples retrieved from IoT projects are:

Automation, Availability, Compatibility, Frequency, Integrity, Interoperability, Mobility, Performance, Precision, Privacy, Range, Reliability, Safety, Scale, Security, Transparency, Trust, Ubiquity, Usability.

Some attributes examples retrieved from IoT projects are: Cost, Power, Size, Weight.



	DONE	TO DO	N/A
Define what are the relevant attributes and their definition for the IoT project (such as "the voice command should always be available" - related to Availability, The ability of the service to be always available, regardless of hardware, software, or user fault).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define what are the measures and metrics for the selected attributes (Measure for Availability = uptime ÷ (uptime + downtime)).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe the implementation mechanisms for the selected attributes (such as using redundant infrastructure components for the voice command to ensure availability).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe the observation and testing mechanisms for the selected attributes (such as the voice command will be tested monthly and should have 99% available time).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



THINGS

1. Define and Implement components. As a starting point in the IoT concept definition, considering the goals established from the problem domain is possible to extract the components required to achieve such a goal. After the components are identified, they all need to be defined with more detailed descriptions. After the components are identified and define, the components need to be implemented in the system. It is an ongoing activity, as the items should be revisited considering new information that can update the implementation, such as the environmental influence on a given component.



	DONE	TO DO	N/A
Define components attributes (such as power, size, and memory).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe the component's behavior (such as actuation, identification, monitoring, and sensing).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify external partners (not internal to the system but are required for the solution).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify components for data exhibition (such as dashboard solutions and applications running in the user smartphone).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish criteria for component selection (such as costs and restrictions).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish component's aim (such as reduce human intervention, track vehicles, connected to the problem domain).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe a strategy for implementing and implementing necessary components (such as using microcontrollers like Arduino and Raspberry Pi, since they can provide a user-friendly development environment).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe a strategy for adapting and adapt necessary components (such as wearables and aid for older adults that should be adjusted to the end-user).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe a strategy for user customization (such as do-it-yourself philosophy using low-cost hardware and 3D printed parts).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



2. Define strategy for integration. It can be a software layer (like middleware) or a physical layer (like circuit adapters) or another alternative that lies between the system components on each side as a bridge.

	DONE	TO DO	N/A
Describe and implement a strategy to address integration issues (such as using interoperable open standards).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe a strategy for integrating necessary components (such as using modular or layered architecture).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify heterogeneity and incompatibility issues among components (such as check the communication technologies and protocols in use).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

3. Define device protection. Related to the physical integrity of the components (system safety), like calibrate power. It is a responsibility to keep the system a state safe and not in danger or at risk (Cambridge Dictionary).

	DONE	TO DO	N/A
Define physical and external threats among components (such as fires or flooding in the location of the components).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define the component's holder and integration needs (such as a combination of modules, a socket, or a device).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe mechanisms to mitigate the threats (such as security measures to prevent physical damages).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe and implement a strategy to address threats (such as Parental Control where changes can be made only by authorized individuals).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

4. Implement components identity. From our IoT definition, the object should be uniquely identified, addressable. It should provide all the device identity information.

	DONE	TO DO	N/A
Define management procedures (such as how to add or remove, enable or disable components from the system).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe and implement device authentication (such as access control, to ensure that the system verifies the credentials).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe and implement device identity (such as by IP address, with attributes and metadata defining physical or virtual identity).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



BEHAVIOR

1. Define identification. The behavior of identifying things by labeling and enabling them to have an identity, recover (through reading), and broadcast information related to the thing and its state. It refers to physical identification - when objects are tagged with electronic tags containing specific information, making it possible to identify objects through tag readers. Not to be virtually identifiable in the sense of connectivity (e.g., IP address).



	DONE	TO DO	N/A
Define the object to be identified (can be a car, a product or a person for example)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define the metadata related to the object (id, name, and description for example)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define an identification technology (QR code or RFID, for example)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe the reading event (Manual or automatic, for example)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe the type of identification (Static/Movel, Active/Passive, Disposable...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

2. Define sensing. The primary function is to sense environment information, requiring information aggregation, data processing and transmission, controlling external context. Enables awareness, thus acting as a bridge between the physical and digital world.



	DONE	TO DO	N/A
Define the data related to the sensing (data to be extracted by the sensors with syntactic and semantic meaning...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe a response for abnormal conditions (send an alert, activate actuation...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Indicate the desired threshold and values (normal condition, safe values....)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify the sensing device (pressure , temperature sensor, Motion sensor...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the sensing rules (schedule-based sensing, event-based sensing, always-on sensing...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



3. Define actuation. Mechanical interventions in the real world according to decisions based on aggregated data or even upon actors' right trigger; relay on responses to the collected information to perform actions in the physical world and change the object state.



	DONE	TO DO	N/A
Describe the manual or automatic mode (the use of rules, threshold, or response time can be applicable. In a smart farm, the irrigation is automatic according to the temperature.)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Locate the action (In a smart farm, release water in the farm)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify who triggers the action (device or human user, for example. In a smart farm, the farmer triggers the action.)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Indicate the circumstances for triggering action - input (if the sensed date is below what is expected according to a defined threshold. In a smart farm, the irrigation is automatic when it is above 30 degrees C)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the consequences of an action - output (In a smart farm, the farm is irrigated)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

4. Define monitoring. A solution to watch, keep track of, or constantly check for a special purpose (observing without control).



	DONE	TO DO	N/A
Define data to be monitored (environmental or heathy information for example)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe monitoring rules (such as where to send information, alerts, and flags)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify the monitoring device (sensor, tag...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Indicate the monitoring temporality (real-time, once a day, during summer...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



5. Define data exhibition. Elements that consume data for exhibition purposes, including the devices that enable data visualization.



	DONE	TO DO	N/A
Define data to be exhibited (make human sense from the data received, graphs, dashboards...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define data-manipulation rules (from what is exhibited, what can the user do)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify exhibition device (mobile app, tv monitor or smart-watch...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



INTERACTIVITY

1. Define involved actors. Identify any human, object or thing that engages in an interaction with the system, including other systems.



	DONE	TO DO	N/A
Define system admin and responsibilities.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define the users, roles and responsibilities (Consider user, business, legal, regulatory and functional issues: for example, requirements for special needs).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define actors' control over the system.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



CONNECTIVITY

1. Establish Connectivity. For dynamic linking IoT services is necessary compatible connectivity based on topology, architecture, constraints and standards.

	DONE	TO DO	N/A
Define network topology and architecture (such as how the connection is organized, node-to-node communication through two active devices with NFC).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define connectivity constraints (Considering the systems requirements, define connectivity constraints such as frequency, range, nodes, power, data rate).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define and Implement connectivity standards (to enable the system to operate sharing the same environment by using LoWPAN , BTv5 or ZigBee...).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish Service Discovery mechanisms. Different IoT solutions can require different properties to identify suitable services to mash-up (for example, use semantic-based similarity and quality of service).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



SMARTNESS

1. Define and Implement smartness. Smartness deals with the combination of characteristics that enable the IoT system to be semi or entirely autonomous for performing any action in the environment. The actions are associated with the smartness ability, depending on the application domain and the user’s needs. These characteristics of smartness are systems requirements. The data collected from the environment support the system to be aware, decide, and act. Therefore smartness should be defined according to user’s need, combining **Environment, Data and Behavior** facets.

	DONE	TO DO	N/A
Describe and Implement AI technology (ex. Machine learning, Fuzzy logic).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe and implement data processing (ex. Define what kind of analysis is required, such as interpretation or management).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe and implement data semantics (ex. data interpretation and ontologies).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe a strategy for real-time operation (Real-time decision, real-time monitoring, or real-time visualization).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify the decision-makers (ex. users, software system).	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



DATA

1. Define and Implement the data model. The project should provide the modeling of the data sources in the system definition. It should capture the relationships existing between a source and the physical environment and the relationships existing among data sources themselves. It is also used to specify the properties and characteristics of the retrieved data. Data has great value for the IoT systems, and it is as relevant as the sources defined for the project in question. In a world of possibilities, it is necessary precision and adequacy to determine relevant data.



	DONE	TO DO	N/A
Define the properties (such as metadata, data types required to achieve the tasks at hand)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define what data will be collected from the components, users and external systems (volume, variety, speed, value, static, dynamic)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Define the inputs and outputs applicable for each data source (assure that the devices or systems are getting and generating accurate data)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe the relationships and flows (such as express relations among the components)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe how data will be used (related to the system behavior and the goals defined in the problem domain)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe how data will be collected from each source (such as cloud, Bluetooth...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe how data will be shared (such as cloud, Bluetooth...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe and implement the data model (Conceptual, Physical and Logical data modeling)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identify the data sources (such as sensors, actuators, external partners...)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



ENVIRONMENT

1. Define relevant environment information. The environment where the solution is deployed is a multi-dimensional contextual space with different levels of importance that can change over time. It is necessary to state the contextual variables to be used to translate the environment into computing technologies when considering the context. Systems can adapt their behavior according to the information they receive about the environment or the user's, and this information is the context the systems should be aware of.

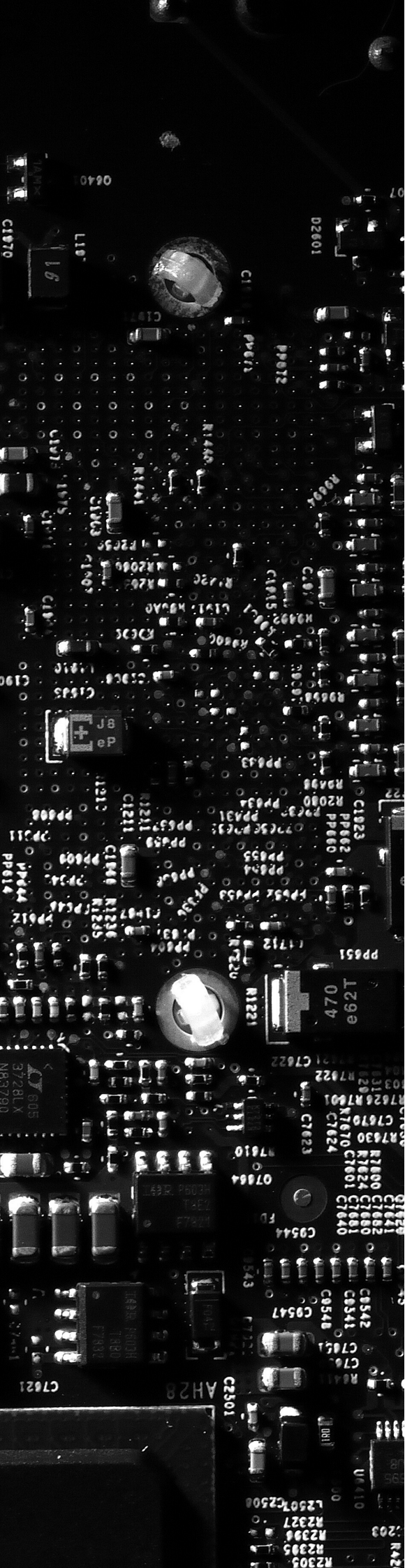


	DONE	TO DO	N/A
Define what data will be collected from the environment (for example, temperature, and humidity)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Describe how to collect data from the environment (for example, using RFID and sensors)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

2. Define the environmental impact. The interplay between environment and solution can have an affect on each other that can alter the desired outcome, which we should be aware of.



	DONE	TO DO	N/A
Describe a strategy for the issues observed (ex. physical protection)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the environmental impact in the solution (physical - ex. deployed in the water)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the environmental impact in the solution through time (physical - ex. when it rains)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the environmental impact on the quality of the solution (virtual - ex. city noise crashing with the building quality)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Establish the solution impact in the environment (physical - ex. birds living nearby)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



By the end of this IoT Roadmap, the project team should be able to have a clear understanding of the problem domain, the operational requirements within, and the conditions and constraints for the solution.

RESULTS

The marks on the roadmap indicates:

ITEMS DONE	ITEMS TO DO	ITEMS NOT APPLICABLE

NEXT STEPS

Depending on the project plan, take the opportunity to revisit Cross Cutting items and make the necessary adjustments in the engineering process.



If the product is already completed, consider handling specific items for the IoT solution, such as:

- Documentation, installation and update procedures (related to maintainability, should include maintenance and replacement orientations).
- Procedures for remote boot and physical boot (for example, how to make a new device registration and the first connection).
- Recovery and contingency plan (it can start by establishing vulnerability action procedures).

CONCLUSION

We hope the Roadmap can contribute to the understanding of IoT and its related information. By using the IoT Roamap to support project decisions to perceive and handle needs, demands and risks associated with engineering an IoT solution.

For questions or suggestions, please contact: rmotta@cos.ufrj.br

Related institutions:



FURTHER INFORMATION

About IoT:

- Motta, R. C.; da Silva, V. M.; Travassos, G. H. "Towards understanding of IoT".
Link: <http://doi.org/10.5753/jserd.2019.14>
- Motta, R. C., de Oliveira, K. M. , Travassos, G. H. "On challenges in engineering IoT systems."
Link: <http://doi.org/10.5753/jserd.2019.15>
- Motta, R. C., de Oliveira, K. M. ,Travassos, G. H. "A Framework to Support the Engineering of Internet of Things Software Systems."
Link: <http://doi.org/10.1145/3319499.3328239>

About Quality Characteristics:

- ISO/IEC 25000:2014 Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuRE)
Link: <https://www.iso.org/standard/64764.html>

About Supporting Technologies:

- de Souza, B. P., Motta, R. C., Travassos, G. H. "Towards the Representation of Smartness in IoT Scenarios Specification"
Link: <http://doi.org/10.1145/3350768.3351797>
- de Souza, B. P., Motta, R. C., Travassos, G. H. "The first version of SCENARIotCHECK: A Checklist for IoT based Scenarios."
Link: <http://doi.org/10.1145/3350768.3350796>
- da Silva, V. "SCENARIoT Support for Scenario Specification of Internet of Things-Based Software Systems"
Link: <http://zenodo.org/record/4080680>

About Systems Engineering:

- SEBoK contributors, "Guide to the Systems Engineering Body of Knowledge", May 2020.
Link: <https://www.sebokwiki.org/>