

Article

# Integrating Citizen Participation in the Development of New ICT Services for Smart Cities

Alexander Jesus Ricardo <sup>1,\*</sup>, Mónica Ayde Vallejo <sup>1</sup> and José Edinson Aedo <sup>2</sup> 

<sup>1</sup> Departamento de Energía Eléctrica y Automática, Facultad de Minas, Universidad Nacional de Colombia Sede Medellín, Medellín 050001, Colombia; mavallejov@unal.edu.co

<sup>2</sup> Departamento de Ingeniería Electrónica y Telecomunicaciones, Faculty of Engineering, Universidad de Antioquia, Medellín 050010, Colombia; jose.aedo@udea.edu.co

\* Correspondence: ajricardor@unal.edu.co

**Abstract:** The transition of cities towards a smarter approach significantly benefits from citizen participation in the development and implementation of innovative information and communication technology (ICT) products and services. Despite the emergence of various initiatives in recent years aimed at guiding the development of smart cities, there is still a lack of effective strategies to actively engage citizens, businesses, and educational institutions during the creation of these products and services. This study describes a set of practices that includes four co-creation techniques to facilitate the effort of software system development in collaboration with citizens and other stakeholders. The SEMAT standard is used to create and represent a method in which these practices are distributed across four stages: focus, definition, development, and validation. In each stage, a practice is proposed that incorporates a co-creation technique and complementary activities from various software engineering disciplines to promote active citizen participation; stimulate idea generation; and facilitate the creation of necessary documents and components for the development of the desired software system, including design systems, code files, conceptual representations, and technical diagrams, among others. Finally, the applicability and completeness of the method are validated through expert consultation in the fields of software engineering and smart cities. Recognized procedures are followed to obtain qualitative and quantitative results, such as improvement actions (addition or removal of elements), levels of consensus or acceptance, and opportunities for future work.

**Keywords:** method; software development; smart cities; citizen participation; value co-creation



**Citation:** Ricardo, A.J.; Vallejo, M.A.; Aedo, J.E. Integrating Citizen Participation in the Development of New ICT Services for Smart Cities. *Information* **2024**, *15*, 812. <https://doi.org/10.3390/info15120812/>

Academic Editor: Enrico Denti

Received: 5 November 2024

Revised: 3 December 2024

Accepted: 4 December 2024

Published: 17 December 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The concept of a citizen-centered smart city refers to the 3.0 model of a smart city. This model emphasizes citizen empowerment as a fundamental element in the transition of cities towards more intelligent and sustainable urban environments [1,2]. Such empowerment enables the delegation of power by governmental entities and the integration of citizens to take control over decisions that affect their community.

In this work, we adopt the approach of Nam and Pardo as a widely recognized definition of the concept of a citizen-centered smart city. The authors describe this concept through three interconnected dimensions: citizenship, technology, and governmental entities [3]. These three dimensions form a “collaborative arena where citizens, governmental entities, and other stakeholders contribute to the development, implementation, and use of intelligent and sustainable technological tools, policies, products, and ICT services”. The goal is to improve public services, optimize resource consumption, enhance mobility conditions, reduce security incidents, optimize energy use, and generally improve citizens’ quality of life [1,4,5].

In the context of citizen-centered smart cities, numerous ICT products and services are implemented, ranging from sensors and communication devices to artificial intelligence models and automated systems. User-focused software systems are frequently employed

to address the challenges faced by citizens in areas such as security, mobility, environment, citizen participation, and public infrastructure within a smart city. However, developing these systems in the context of citizen-centered smart cities faces additional challenges compared to conventional or business projects. These challenges arise due to the need for collaboration and active participation of multiple stakeholders; the lack of prior documentation; the social nature of the problems; and the diversity of stakeholders with their respective social, political, and cultural differences [6–11].

Some authors highlight that in citizen-centered smart cities, the success of developing and using user-focused software systems heavily depends on the level of citizen participation. This participation facilitates the identification of real needs by stakeholders from the early stages of the endeavor. Similarly, it enables direct validation of developed prototypes and ensures that proposed solutions will positively impact end-users [12]. This article considers the discourse initiated by Cardullo and Kitchin in 2019 [13], who address the different levels of citizen participation in a smart city. In this proposal, the concept of “the scaffold of smart citizen participation” is introduced to analyze whether the initiatives developed in a city are citizen-centric. This work proposes value co-creation as a strategy to articulate citizen participation, focusing on the higher levels of participation within this framework, as citizen power is considered a key element in the decision-making and management processes associated with the definition and implementation of new ICT services in a city.

Conventional methods and frameworks, such as RUP, AUP, XP, or SCRUM, are widely recognized in the software industry. These methods contain well-defined activities, roles, and work products, providing a foundational structure that facilitates software system development efforts in various business scenarios [14,15]. However, these alternatives alone do not meet the needs of citizen-centered smart city contexts, especially regarding the participation and integration of citizens and other stakeholders in the territory [7].

The alternatives mentioned outline development phases covering various software engineering disciplines, including business modeling, requirements engineering, analysis, design, and implementation [16]. However, these methodologies do not account for the complexity of identifying needs in contexts involving multiple stakeholders. They also lack participatory strategies tailored to the social, cultural, and political diversity inherent in citizen-centered smart cities [17]. While these frameworks detail the required work products, they fail to provide clear mechanisms for engaging citizens and other territorial actors in their creation.

Hybrid approaches [18,19] attempt to incorporate participatory elements and combine practices from robust and agile methods. Nevertheless, they fall short of providing concrete guidelines for managing the scale and diversity of involved stakeholders, as well as standardized structures that help development teams identify the necessary activities, work products, and profiles to carry them out effectively [20].

Despite the existence of various alternatives in the literature to address the challenges of development within the context of citizen-centered smart cities, most focus on case studies testing empirical techniques based on citizen participation mechanisms, for example, the experiences described in [9,10]. In other cases, authors propose processes that incorporate activities distributed across iterative agile cycles [21]; however, no proposals have been identified that combine the formality of a method—including the use of standards, well-defined activities, required roles, and technical and functional work products, with participatory techniques that facilitate the integration of citizens and other territorial actors.

As a result, recent initiatives aiming to develop user-oriented software systems have turned into unconventional alternatives or hybrid processes [22]. These new approaches seek to adapt existing work methods to better align with the needs of citizen-centered smart cities [18]. The current challenge in this context lies in defining processes and strategies that can effectively guide citizen and stakeholder participation throughout all phases of system development—from identifying needs to implementing and maintaining the developed solutions [21,23–25].

A process for software systems development in the context of citizen-centered smart cities should facilitate the participation of citizens and relevant stakeholders through didactic strategies that promote teamwork and idea generation to identify needs, prioritize requirements, design solutions, and validate developed systems [26]. Additionally, the process should incorporate essential activities to develop the work products that shape the software system, including code files, architecture diagrams, database schemas, design systems, and more. Moreover, it is crucial for the process to include complementary activities to identify stakeholders and key partners (public entities, private organizations, and educational institutions) that can add value to the process. Finally, the process should support continuous feedback from citizens and relevant stakeholders, implying the implementation of iterative and incremental structures that allow for adjustments to the work products as activities progress [6,27,28].

This study presents a method as a practice set that includes various co-creation techniques as a strategy for citizen participation and a set of activities, roles, competencies, and work products that shape software systems within the context of citizen-centered smart cities. Practices are framed in an iterative and incremental process of four phases to guide the effort, from identifying citizens and relevant stakeholders to validating the developed systems.

Our proposal does not aim to reinvent existing methods and practices to comprehensively address the software lifecycle in business contexts. The goal is to complement these practices with activities and work products that address the specific conditions of citizen-centered smart cities. To achieve this, we base our approach on RUP, a robust and widely recognized method that includes activities for thoroughly developing software system components. Additionally, we incorporate concepts from Design Thinking, a framework focused on user collaboration through didactic tools that facilitate idea generation, prototyping, and participatory validation.

Finally, we use the core of the SEMAT essence as a standard for defining and representing the practices that comprise the method and all its components (activities, roles, competencies, and work products). SEMAT also guides the development of schemas and specification tables that facilitate understanding the method's structure and the elements necessary for carrying out activities and creating work products [29].

This article is structured as follows: Section 2 provides background on the development of ICT products and services within the context of citizen-centered smart cities; Section 3 describes the research development process; Section 4 analyzes the evolution of the concept of citizen participation and its relationship with the development of ICT products and services in smart cities; Section 5 details the proposed method as an alternative to mitigate the identified problem; and finally, Sections 6 and 7 present the validation process, the results obtained, and the conclusions of the work conducted.

## 2. Background

Several initiatives highlight various techniques and procedures to facilitate active citizen participation in the development of ICT products and services within the analyzed context. Four main approaches can be identified: methodological frameworks, development platforms, support tools, and use cases [30].

This work focuses on methodological frameworks aimed at the development of ICT products and services through participation techniques based on face-to-face interactions, applicable in different contexts of a smart city. It also considers use cases that describe specific experiences with empirical procedures applied in particular areas of a smart city. Table 1 presents several related studies that describe some approaches.

As observed in Table 1, there are publications that describe solutions to the problem of developing ICT products or services in the context of citizen-centered smart cities, most of which present case studies showing promising results. However, upon detailed analysis, it is evident that the proposed solutions are generally described and lack the necessary specification to be adopted as guides for developing software systems in the analyzed

context (smart city 3.0). Furthermore, they are oriented towards specific problems and need adaptation to be used in other contexts.

**Table 1.** Related work.

Title	Summary
Living Lab: an open and citizen-centric approach for innovation [31].	The authors describe a three-phase cyclical process: Concept Design, Prototype Design, and Final System Design. Each cycle aims to understand citizen needs, generate innovative solutions, and develop services that address these needs.
Viewpoint participation technique: A model of participative requirements elicitation [32].	A model is proposed for requirements elicitation through stakeholder participation. The model addresses the problem of identifying requirements in complex contexts by means of activities that help understand the characteristics of the problems, current solutions, and their relationship with the stakeholders' context. Additionally, it suggests working with all stakeholders to gather different perspectives and understand the problem domain, so that requirements can be developed to describe a potential solution.
System engineering for smart cities-hybrid-agile approach in smart cities procurement [18].	A hybrid method is proposed for software development in the context of smart cities. The method uses the SCRUM framework as a foundation, modifies its role structure, and provides recommendations for adapting this agile framework to the context of smart cities. The benefits of stakeholder participation in the process are highlighted, as well as the weaknesses of standardized methods for software development in the smart city context.
Co-creating with consumers and stakeholders to understand the benefit of Internet of Things in Smart Living Environments for Ageing Well: The approach adopted in the Madrid Deployment Site of the ACTIVAGE Large Scale Pilot [33].	A framework for the co-creation of IoT infrastructure is proposed. It describes a two-iteration process in which user needs are gathered based on prior experience and existing literature. Subsequently, use cases and scenarios are created to define a solution based on a combination of services and technological infrastructure. The solution is shaped by the available service providers. In the second iteration, the solution is made available to stakeholders, and the benefits of its implementation are analyzed.
Guidelines Towards Better Participation of Older Adults in Software Development Processes using a new SPIRAL Method and Participatory Approach [21].	A method based on the living labs strategy is proposed for developing software with older adults as stakeholders. SPIRAL is defined as a rapid and agile process consisting of four steps or stages. In the first stage, technological barriers and the skills of older adults for working with ICT products are identified, using previously built digital platforms. In the second stage, stakeholder (older adults) involvement is encouraged through mobile devices and applications to familiarize them with the technologies to be used. In the third stage, design activities, requirements gathering, and rapid development cycles are carried out through Hackathon activities. Finally, in the fourth stage, stakeholder participation is promoted to tailor the results to their needs through co-design activities.
A Scrum-Based Development Process to Support Co-creation with Elders in the {eHealth} Domain [7].	A process based on SCRUM is proposed with four stages: evaluation, co-creation, prototyping, requirements definition, and testing. The use of focus groups, contextual walkthroughs, and usability testing is suggested to facilitate citizen participation.
Co-creation in Practice I: Co-creating a Digital Neighbourhood Guide (Bremen Osterholz) [9].	Co-creation techniques such as surveys and participatory design workshops are introduced. Focus groups and feedback meetings are used to adjust the guidelines based on user responses.
Co-Creation in Practice (II): Co-creating a Digital Walking Guide (Bremen Hemelingen) [10].	Co-creation techniques, including contextual walkthroughs and collaborative workshops, are recommended. Participation is encouraged through an online platform to capture citizen suggestions.

According to Simonofski and others, the trend in software development for citizen-centered smart cities is the implementation of structured and well-documented processes that guide the process from need identification to system deployment. Recent literature reviews, such as [14] and that of Simonofski [15], indicate that traditional, agile, or non-agile methods and frameworks are the most utilized; among these, RUP, XP, SCRUM, AUP, or waterfall development stand out.

Based on some experiences described in these reviews, smart city project developers and managers focused on software system development agree that the most appropriate

approach is to use elements from various available methods to address the inherent difficulties of development in the analyzed context. These approaches are known as hybrids [15]. This implies combining and modifying existing practices to adapt them to the conditions of citizen-centered smart cities.

Among these reviews, one of the publications described in Table 1 emerges as a hybrid alternative for developing software systems in the context of smart cities. This study [18] is considered one of the closest contributions to the solution presented in this work, but it lacks defined strategies to facilitate stakeholder participation in constructing the necessary work products and focuses on defining roles that facilitate direct understanding with citizens and other stakeholders.

Another significant contribution is described in [21]. Although it is exclusively oriented towards working with older adults, it proposes four phases with innovative activities combining participation strategies such as hackathons, co-design techniques, and living labs to facilitate the development of software systems in conjunction with stakeholders (older adults). While this proposal can be adapted to work with other types of stakeholders, it lacks the definition of the work products to be developed and a complete description of the activities to be performed. Therefore, definitions associated with creating code files, requirement specification, functionality prioritization, and how citizens can add value in different phases of the process are left out.

Finally, there is a trend toward creating or adapting processes and strategies to guide the design and development of ICT products and services in smart cities. However, there is still a lack of detailed procedure that public, private, or mixed entities can use to facilitate citizen participation in all phases of software system development.

### 3. Methodology

To achieve the proposed objectives, a process for defining theories in the field of software engineering is used as a foundation. The methodology involves two phases: construction and validation, to shape the method for developing software systems in the context of citizen-centered smart cities. The phases and activities are described below [34].

#### 3.1. Construction

The construction of the method and other complementary elements involves the following activities:

- Define mechanisms for stakeholder participation and integration: These mechanisms are defined based on the results of a literature review related to strategies for facilitating citizen participation in the context of smart cities. They are selected and adapted according to the expected work products at each phase of the method.
- Build the method for developing software in the context of smart cities: This activity involves integrating the mechanisms for citizen participation, roles, work products, and other elements that make up the method.
- Create schemes and conceptual representations of the method: This activity involves developing the necessary schemes and representations to present the method in a standardized way and facilitate its understanding. The schemes visually present all elements of the method and the proposed execution flow.

#### 3.2. Validation

The validation phase is carried out using the process proposed by [35]. The authors describe the steps to plan and conduct focus group sessions with experts in the relevant field. This allows for obtaining suggestions and quantitative data to improve the method and validate the achievement of the proposed objectives.

- Plan the validation: In this activity, the objectives of the validation are defined, and the necessary inputs to carry out the process are prepared.

- Define the experts: This activity involves defining the profile of the experts participating in the validation and extending invitations to form a group of three experts who meet the required profile.
- Conduct the validation: In this activity, the focus group session is conducted with the defined structure and inputs.

#### 4. Value Co-Creation in a Citizen-Centered Smart City Context

The use of the concept of citizen participation became evident during the social movements in the 1960s, when university students and workers demanded to be considered in decision-making processes for the formulation of public policies. Sherry Arnstein's ladder of participation was one of the earliest attempts to classify and categorize citizen participation in urban planning and development processes [36]. The participation ladder was conceptualized as a model for describing different levels of participation, ranging from manipulation and control by authorities to genuine citizen participation and empowerment.

From Arnstein's study, research emerged aimed at proposing strategies for transitioning from an ideal concept of participation to a practical approach applicable in real-world scenarios. This was done to counter the tendency of governmental entities to use citizen participation as a mechanism to justify decision making regarding the creation and implementation of projects and public policies [37].

The role of citizens in smart cities is far from uniform and often reflects significant geographical variations. For instance, as noted by Burns and others in 2021, citizen participation in smart cities in Northern Europe often emphasizes collaborative governance, while in the Global South, these initiatives may focus on addressing infrastructure deficits or socio-economic inclusion [38,39]. This suggests that, although smart city initiatives are developed within a shared technological and conceptual framework and may have global reach and impact, their effectiveness is maximized when adapted to the specific conditions of each territory [25]. In this way, the designed solutions are more likely to be accepted and used by citizens, as they directly address their real needs.

In this context, the concept of citizen science emerges, referring to the active involvement of communities in the collection, monitoring, and analysis of scientific data, often in collaboration with professionals or under the guidance of scientists [40,41]. This approach broadens the traditional role of citizens in research, allowing them to contribute to knowledge creation and problem solving through various methodologies and technologies. Citizen science has a significant impact both locally and globally, as it enables communities to actively participate in generating innovative solutions. Furthermore, this approach has diversified into forms such as do-it-yourself science, crowd science, and participatory monitoring, all of which empower citizens to contribute to long-term decision making and strengthen community resilience [42,43].

Citizen participation in smart cities and citizen science are, therefore, complementary approaches that seek to engage citizens not only in urban decision making but also in scientific problem solving. While citizen participation in smart cities focuses on enabling citizens to influence public policies that affect their daily lives, citizen science goes a step further by involving citizens in the process of generating data and scientific knowledge to address urban and global challenges. Both approaches align with the goal of improving the effectiveness of public policies and fostering more inclusive governance.

Expanding on Arnstein's ideas, Weber and others in 2019 organized the citizen science concept into four levels: crowdsourcing, distributed intelligence, participatory science, and extreme citizen science. These levels vary in the depth of citizen involvement, from simple data collection in crowdsourcing to full project leadership in extreme citizen science, where participants design and execute the entire project without professional scientific guidance. This classification highlights the diversity of scientific participation practices, ranging from basic contributions to the complete management of research projects [44].

On the other hand, citizen participation can be classified into two groups: passive and active participation. According to the conceptual model described in [45], citizens engage

in passive participation using smart public products and services equipped to capture and analyze data. These systems employ algorithms based on technologies such as big data or machine learning [37].

Passive participation promotes continuous improvement of public services through the analysis and processing of captured data, albeit requiring suitable technological infrastructure, which is typically lacking in cities transitioning toward smarter urban spaces [46].

Conversely, active participation is based on continuous feedback from citizens who actively contribute to the construction and utilization of products and services that benefit them. In this context, active participation fosters technological and urban growth, which serves as a gateway to passive participation through the entire built technological infrastructure [23,47].

Recent advances in information and communication technologies (ICT), particularly mobile applications, have created ideal conditions to enhance citizen participation and citizen science processes in cities. These technologies enable dialogue between citizens, scientists, and policymakers in virtual spaces, facilitating data collection, analysis, and evidence-based participatory decision making. These tools not only promote collaboration but also optimize the implementation of data-driven urban policies, contributing to a more transparent and effective model of governance [48].

In this paper, we focus on active participation and collaboration between citizens and other stakeholders in the territory, under the guidance of professional experts in the creation and implementation of ICT products and services. In this way, we draw on relevant aspects of the two concepts previously described to form a solid conceptual foundation that supports the development of a solution to the problem of designing and developing software systems as ICT products or services in the context of smart cities.

In the literature, there are various practices associated with citizen participation and value generation through the interaction between different actors [49,50]. Co-creation, co-design, and co-production are some of the most recognized concepts and refer to the collaborative generation of knowledge between academics and stakeholders from other sectors, including service users, caregivers, professionals, and commissioners. The term “co-production” was coined by Elinor Ostrom and her colleagues in the 1970s to refer to citizen participation in the production of public services [51], co-creation has its roots in marketing and business literature [52], and co-design originated in the participatory design movement in Scandinavia in the 1970s [53]. Co-creation focuses on creating value through active collaboration between multiple public and private actors, as well as a constructive exchange of different types of knowledge, skills, ideas, and resources for public planning, problem solving, and policymaking [54,55], whereas co-production refers to citizen participation in the implementation of public services.

Value co-creation is a highly useful tool when aiming to address citizens’ problems in any of the critical areas of a smart city, and there is a lack of information, documents, or resources that would allow for directly moving to the design or implementation of ICT products or services [24]. According to [56], existing co-creation techniques can be grouped based on their purpose: research, team building, ideation, development, definition, and validation. The combination of techniques from all these groups contributes to the creation of additional comprehensive strategies that address all aspects of a smart city initiative.

Several co-creation techniques are extensively used in the creation of ICT products and services in smart cities. These include participatory design workshops, collaborative mapping, open data platforms, mobile applications, and prototyping workshops [21,24,57,58].

These strategies are complemented by approaches such as living labs, which enable the development, testing, and validation of technological solutions through direct interactions with technology and educational tools. Together, these techniques promote greater citizen participation and a more efficient collaborative approach to designing and developing technological solutions for smart cities [59].

In this context, this proposal utilizes value co-creation as a strategy for citizen participation. Existing co-creation techniques are used as a basis and modified with the aim of

integrating them into the development process, thus facilitating the definition of needs and requirements, the prioritization of functionalities, and the validation of the developed systems. Some of the suggested techniques incorporate elements from various approaches in innovation theories or participatory design, as well as making use of educational materials such as cards, boards, markers, and others to facilitate the participation of citizens and other stakeholders.

The following section provides a comprehensive description of the suggested techniques, including the required inputs, expected work products, and how they are integrated into the practices that form the method.

### 5. Method Description

According to the definitions adopted from the Essence Kernel, a method is “the way a team conducts its work in a software development effort”. More specifically, a method is a defined set of practices that form a standard and reusable way of working, which can be used in a specific context [29].

As shown in the diagram in Figure 1, the proposed method is an iterative and incremental process consisting of four practices distributed across four phases (focus, definition, development, and validation). As can be observed, the phases are arranged clockwise, starting with the focus phase and ending with the validation phase. In this way, as the activities of each practice are carried out, the work products that enable progress in the development effort are progressively built. In this context, a work product is an “artifact of value and relevance to a software engineering effort” [29]. Work products can be pieces of code, database diagrams, or requirement documents, to name a few examples.

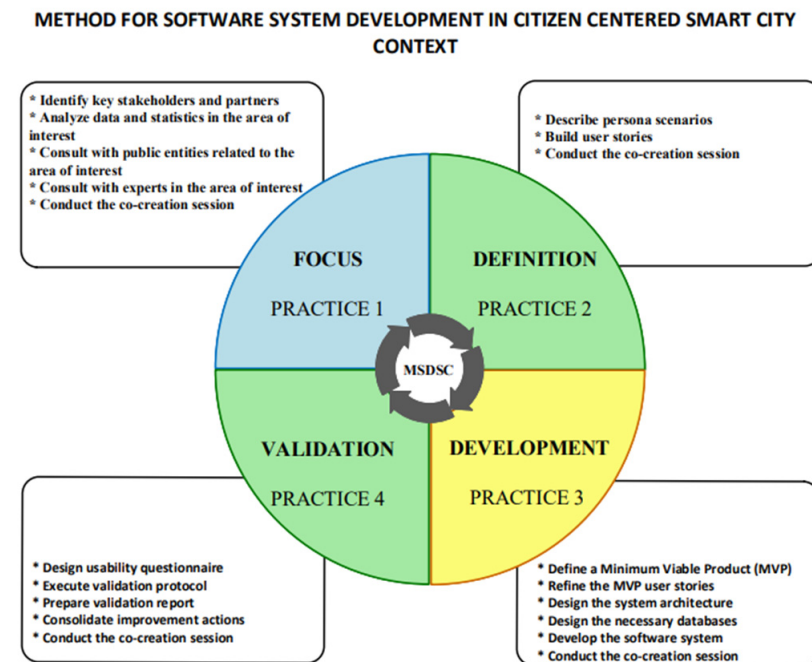


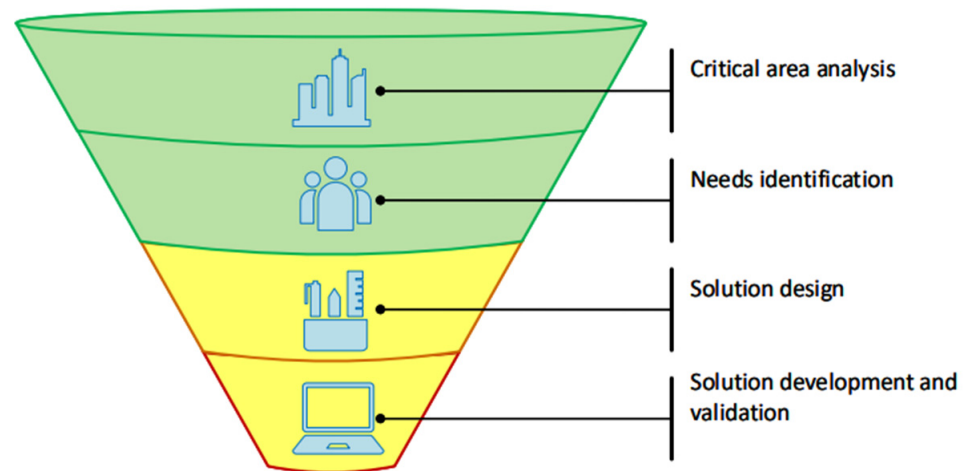
Figure 1. Method schema.

The method addresses the context of smart cities from the general to the specific. As presented in Figure 2, the critical area of interest is analyzed based on existing information, expert consultations, or visits to companies and government entities. The objective of this analysis is to understand the generalities of the contexts in which problems or difficulties affecting the citizens are identified.

Once possible contexts are identified from an external perspective, we work with stakeholders to prioritize their needs concerning the critical area of interest and the previously identified contexts. Stakeholders include all affected citizens; members of private



and public companies; and educational institutions that, in some way, relate to the critical area of interest.



**Figure 2.** Method structure.

Based on the work conducted with stakeholders, the functionalities and technical characteristics of the software system are established. For this purpose, technical and functional specification documents are created, such as requirements, databases, frameworks, platforms, and other inputs needed to develop the software system. Finally, the software system is developed and validated in collaboration with the stakeholders to ensure that the included functionalities meet their needs.

To carry out the described activities, we defined the profiles that make up the work team and a set of associated responsibilities, which facilitate the tracking of activities and validation of the produced work products. The following roles are considered: assistant, coordinator, analyst, designer, architect, developer, tester, manager, and stakeholder.

The assistant possesses competencies such as communication skills, empathy, and creativity. They participate in preparing the necessary inputs for executing activities and support the work sessions by distributing and collecting forms or resolving questions and issues.

The roles of analyst, designer, architect, developer, and tester make up the development team, which has the technical knowledge necessary to create design documents and specifications, as well as user interfaces and code files. The development team is responsible for designing and developing the software system based on the needs of the stakeholders. This team works with the support of the coordinator, who carries out transversal activities to resolve impediments, prioritize activities, schedule work sessions, and control tools and monitor activities.

The manager is knowledgeable in the critical area of interest and capable of making decisions. They are responsible for managing the financial, human, and technical resources necessary to carry out the activities. Additionally, they perform coordination and leadership tasks to support the development team director.

The stakeholder is a citizen from the affected community or a member of a private company, government entity, or educational institution. The stakeholder supports the processes of identifying, defining, prioritizing, and validating the functionalities included in the prototype through participation in work sessions.

### 5.1. Practice 1—Preliminary Research of the Stakeholder Domain

This practice outlines the approach to gain a detailed understanding of the contexts in which citizens perceive problems or difficulties. As depicted in the formal representation in the Essence Kernel (see Figure 3), this practice is entirely conducted during the focusing

phase, with the work products created by team members in the roles of manager, assistant, and stakeholder.

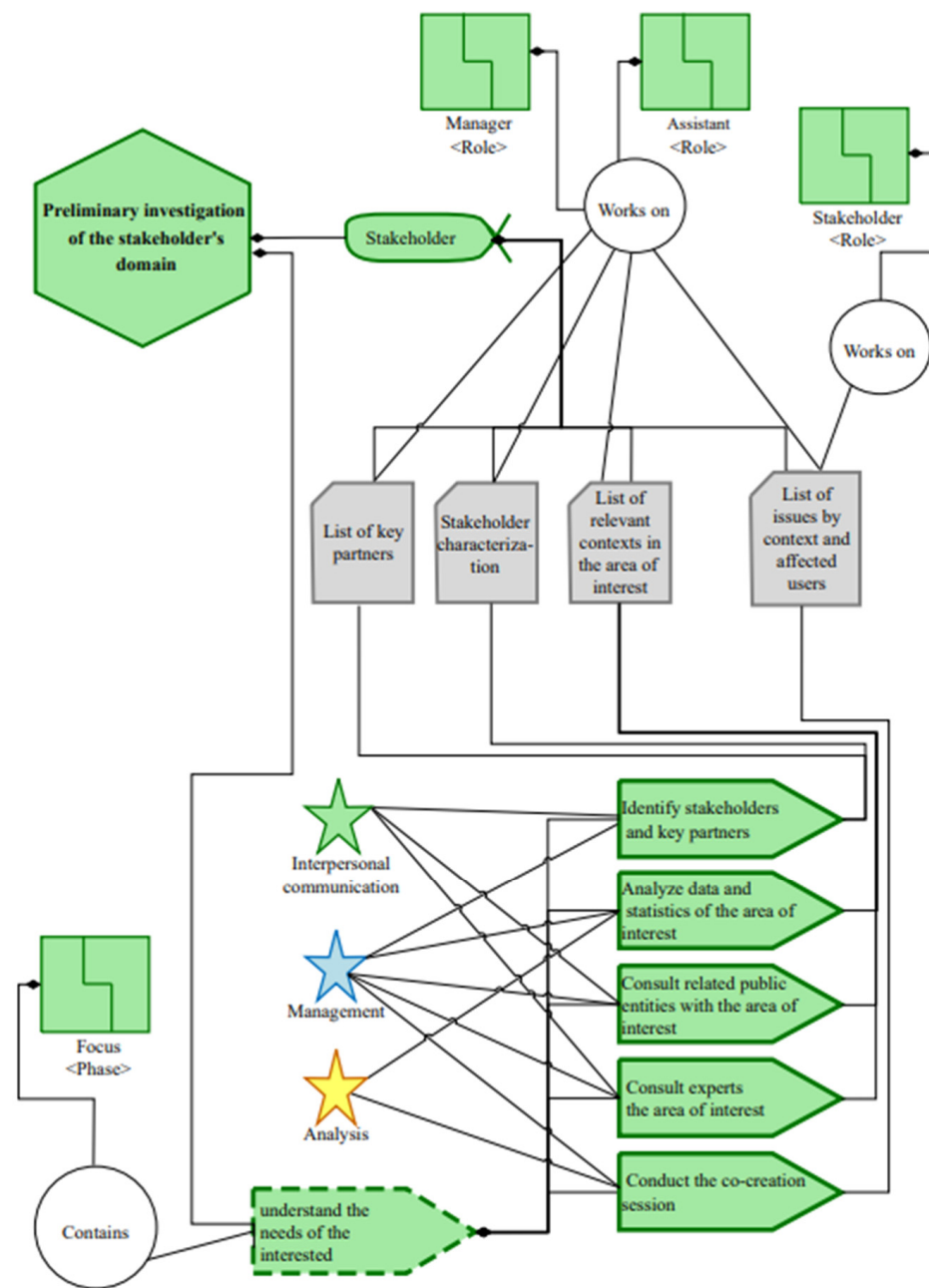


Figure 3. Practice 1—essence representation.


Figure 4 presents the specification card for this practice. The entry criteria are tied to the smart city initiative or project. It is assumed that a viable public, private, or mixed initiative focused on solving citizens’ problems through software systems is in place.

The entry criteria include the necessary work products for executing the practice. This requires a document describing and justifying the smart city initiative or project, as well as documents detailing the critical area of interest, whether it be security, health, transportation, environment, or any other area depending on the identified needs.

The exit criteria are associated with the stakeholders and requirements alphas. By the end of the practice, the stakeholder alpha will be in the involved state, and the requirement’s alpha will be in the conceived state. This is reflected in the associated work products, as a

list of key partners is created, stakeholders are characterized, and contexts and associated problems are identified.

The proposed activities in this practice fall under the activity space “Understand Stakeholder Needs”, focusing on identifying stakeholders and understanding their needs based on existing data and direct interaction with experts and government entities. The activities are described in the order of execution presented in the formal diagram.

	<b>Name:</b> Preliminary Domain Research of the Stakeholder
<b>Description:</b> This practice guides the approach for delving into the stakeholder's domain.	
<b>Entry Criteria</b>	
(Initiative: Viable)	
<b>Work Products Associated with the Entry Criteria</b>	
1 Document describing and justifying the public, private, or mixed initiative	
2 Partial description of the critical area of interest	
<b>Completion Criteria</b>	
(Stakeholders: Involved) (Requirements: Conceived)	
<b>Work Products Associated with the Completion Criteria</b>	
1 List of key partners	
2 Stakeholder characterization	
3 List of relevant contexts	
4 List of issues by context and affected users	

**Figure 4.** Practice 1—specification card.

#### 5.1.1. Identify Stakeholders and Key Partners

It is essential to identify all individuals who can provide knowledge and experience regarding the critical area of interest, including private companies, government entities, educational institutions, and citizens. Previous initiative reports, literature reviews, or entry criteria work products can be used to identify stakeholders and key partners.

This activity results in a detailed list of key partners, including contact information, representative names, and possible contributions (economic, logistical, etc.). Additionally, documents describing the characteristics of the identified stakeholders are generated, including relevant data to form heterogeneous groups for co-creation sessions.

#### 5.1.2. Analyze Data and Statistics of the Area of Interest

Analyzing existing data is important to preliminarily understand the most relevant characteristics of the critical area of interest. These data are generated from surveys or polls that aim to evaluate citizens’ quality of life concerning variables like education level, socioeconomic status, and work form. Descriptive analysis, graphs, clustering techniques, and other methods can be implemented for this purpose.

#### 5.1.3. Consult Public Entities Related to the Area of Interest

Public entities responsible for addressing the needs of the area of interest can provide additional data, reports of completed projects, or other valuable inputs to gain a deeper understanding of citizens’ needs. This can be done through group work meetings, focus groups, interviews, and questionnaires, among others.

#### 5.1.4. Consult Experts in the Area of Interest

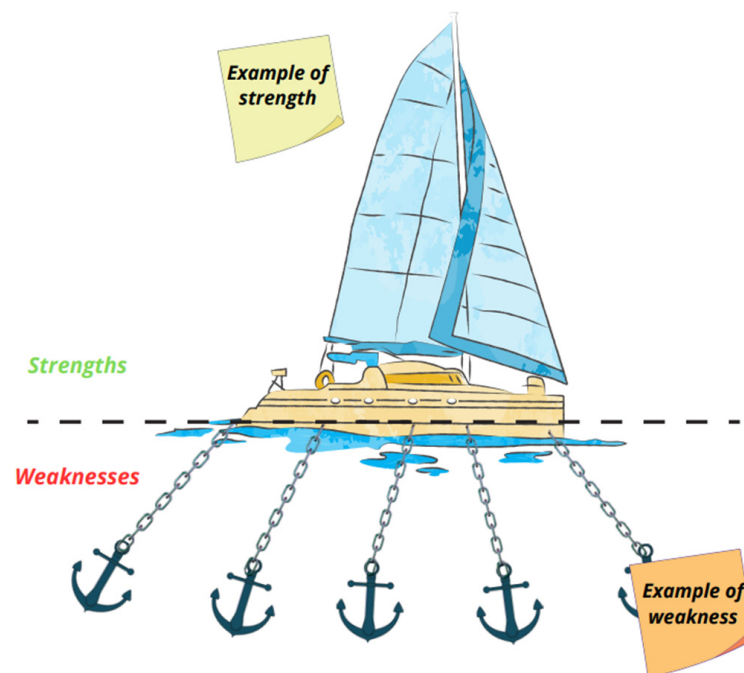
Experts with academic and practical experience in the area of interest can help avoid delays in the process, as they have clarity about reliable information sources and know the procedures to obtain them, including tangible results from previous initiatives, external open-source solutions, and technological platforms, among others.

Together with the previous activities, this results in a list of relevant contexts in interest area, identifying problems that can be improved through software systems. A context groups a set of related problems; for example, in the critical area of security, contexts like urban security, digital security, and home security can be defined. Each context has different problems affecting the same group of citizens in similar scenarios. The context list contains relevant data to select the most important contexts or those with the best relationship between existing problems and feasibility of implementing a solution.

#### 5.1.5. Conduct the Co-Creation Session

In this practice, this activity is carried out using the co-creation technique known as “Anchored Sailboat”. This technique is adapted to identify both strengths and weaknesses and serves as a tool for idea generation by identifying positive and negative events based on the previous experiences of citizens participating in the activity [60].

Participants interact with a representation of a sailboat with multiple anchors (see Figure 5). The anchors symbolize situations that hinder the normal performance of daily tasks, while the top of the sailboat represents positive aspects or exists alternative solutions. Through this dynamic, participants can propose problems and solutions for the analyzed contexts.



**Figure 5.** Anchored Sailboat representation.

Figure 6 shows the specification card for this activity. As can be seen, the entry criteria for this activity are related to the stakeholder alpha, which should be in the recognized state, meaning that the stakeholders are identified, and the corresponding characterization document is created.

For this activity, the sailboat diagram (one per context) should be available in digital or physical format, sized 60 cm × 40 cm (suggested), along with other necessary elements for participants to express group and individual ideas in natural language (markers, sticky notes, sheets of paper, masking tape, or digital tablets, touch screens, electronic pens, etc., depending on the selected format). Below are additional recommendations.

- All participant groups should analyze all contexts (one at a time). This can be done by setting up workstations, allowing participant groups to move through all contexts and review the contributions of previous groups. It is important to note that the contexts are defined from the work product “List of relevant contexts in the area of interest”.

The number of contexts to analyze will depend on the initiative’s scope and the results of previous activities; however, it is recommended to analyze between 3 and 5 contexts per work session.

- The sailboat diagram (physical or digital) should be large enough for participants in each team to write or stick notes with their comments in the appropriate spaces (anchors or sail).
- All participant groups should select the most relevant problems in each context to group and order the problems associated with the same context.
- The results obtained in each context (problems, solutions, and relevance order) should be shared to encourage idea generation and contributions from all participants. This sharing can be done after completing the analysis of all contexts.
- Analyzing the results includes identifying the types of users involved in the identified problems, such as doctors, patients, police officers, criminals, drivers, pedestrians, the elderly, people with disabilities, and students, depending on the critical area of interest and selected context.


	<b>Practice:</b> Preliminary investigation of the stakeholder's domain <b>Activity [5]:</b> Conduct the Co-Creation Session <b>Approach:</b> Preliminary <b>Activity Space:</b> Understand Stakeholder Needs
<b>Description:</b> This activity facilitates the understanding of stakeholder needs through direct interaction with them.	
<b>Entry Criteria</b> (Stakeholders: Recognized)	
<b>Work Products Associated with the Entry Criteria</b>	
1 List of key partners 2 Stakeholder characterization 3 List of relevant contexts	
<b>Tasks</b>	
1 Define work teams. 2 Prepare the necessary technical and educational materials. 3 Set the date, time, and location of the session. 4 Select and invite stakeholders and key partners. 5 Design formats: attendance, informed consent, use of image and data. 6 Introduce the work team. 7 Present the session context. 8 Present the activities and time structure. 9 Present the available technical and educational materials.	10 Organize participants into work teams. 11 Execute the activities with the defined tools and materials for the session. 12 Present the results obtained by participants. 13 Foster discussion of results and formulation of conclusions. 14 Collect results, audiovisual evidence, and informed consent forms. 15 Digitize the obtained information. 16 Synthesize participants' comments. 17 Identify relevant improvements or contributions. 18 Prepare the conclusions and session report.
<b>Completion Criteria</b> (Stakeholders: Involved) (Requirements: Conceived)	
<b>Work Products Associated with the Completion Criteria</b>	
1 List of issues by context and affected users	

Figure 6. Anchored Sailboat activity specification card.

This activity results in the work product “List of problems by context and affected users”, containing the identified problems in the co-creation session, ordered by relevance as defined by the participants and grouped by context. Each problem is associated with one or more types of users affected by the problem or promoting its existence. The types of users emerge from the subsequent results analysis.

5.2. Practice 2—Visual Requirement Management

This practice guides the workflow to identify the functional requirements for the software system to be developed. As shown in the formal representation at the core of the Essence (see Figure 7), this practice is entirely conducted during the definition phase, and the work products are created by team members in the roles of manager, stakeholder, analyst, and assistant.

Figure 8 presents the specification card for this practice. The entry criteria relate to the requirements and stakeholder alphas, which must be in the states of conceived and involved, respectively. This means that the need to develop a software system has been established, and the stakeholders are participating and fulfilling their roles within the team.

The entry criteria are associated with the work products necessary to carry out the practice. As indicated on the specification card in Figure 8, these work products are those created during the focus phase, i.e., in Practice 1: Preliminary Investigation of the Stakeholder’s Domain.

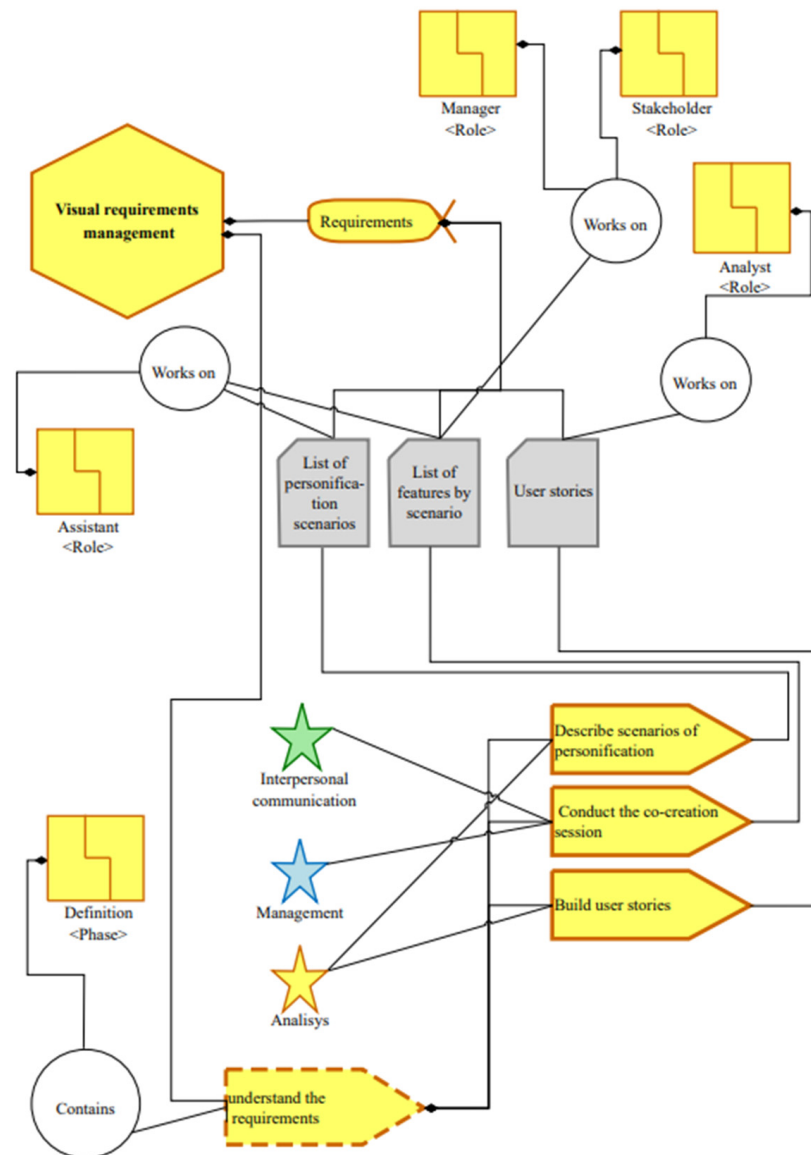



Figure 7. Practice 2—Essence representation.

The completion criterion is associated with the requirement’s alpha. By the end of the practice, the requirement’s alpha will be in the state of coherent, indicating that the requirements describe the essential functionalities of the software system.

In this practice, work products associated with the functional requirements of the software system are created. Based on the problems and user types defined in Practice 1, lists of functionalities are developed to address the identified problems and are subsequently specified in a standardized user story format.

The proposed activities in this practice are grouped in the “Understand Requirements” activity space, focusing on analyzing problems with the stakeholders and defining a set of

functionalities to be included in the software system to be developed. The activities are described below in the order of execution as presented in the formal scheme.

	<b>Name:</b> Visual Identification of Requirements
<b>Description:</b> This practice guides the approach for identifying functional requirements.	
<b>Entry Criteria</b>	
(Requirements: Conceived) (Stakeholders: Involved)	
<b>Work Products Associated with the Entry Criteria</b>	
1 List of key partners 2 Stakeholder characterization 3 List of relevant contexts 4 List of issues by context and affected users	
<b>Completion Criteria</b>	
(Requirements: Coherent)	
<b>Work Products Associated with the Completion Criteria</b>	
1 List of persona scenarios 2 List of functionalities by scenario 3 User stories	

**Figure 8.** Practice 2—Specification card.

### 5.2.1. Describe Persona Scenarios

Persona scenarios are the main input for developing the co-creation session. In this activity, the work product “List of Problems by Context and Affected Users” is used to create scenarios that highlight the selected problems. Below is a specific example:

Critical Area: Public Safety.

Context: Home Safety.

Problem: Violence against Minors.

User Types: Affected Person, Aggressor, Witness to the Crime.

Scenario: You are a neighbor witnessing an act of violence against a minor in your community. What actions would you like to be able to take using an ICT product or service to help the affected minor?


This activity involves creating a list of scenarios for each selected problem. The number of scenarios depends on the user types linked to the problem, considering scenarios from each perspective.

### 5.2.2. Conduct the Co-Creation Session

In this practice, the co-creation session is carried out using the “User Persona” technique. Participants analyze scenarios representing the problems and user types for the selected problems. Each participant “personifies” the described scenario and expresses the actions they would like to be able to take to solve the presented problem from the viewpoint of the assigned user.

This activity fosters idea generation as participants take on the role of the user experiencing a negative event or problem. Figure shows a card-based format that can be used for participants to interact with the problem scenarios and associated user types. Participants describe the actions they would like to take in natural language and associate them with the specific scenario.

Figure 9 presents the specification card for this activity. As shown, the entry criterion is associated with the work product “List of Persona Scenarios”, developed in the previous activity. These scenarios form the basis for preparing the main input for the co-creation session. It is recommended to use a didactic-card-based format to facilitate the stakeholder participation and integration.

	<b>Practice:</b> Visual requirements management <b>Activity [2]:</b> Conduct the Co-Creation Session <b>Approach:</b> Visual <b>Activity Space:</b> Understand Requirements
<b>Description:</b> This activity facilitates the selection of the most relevant requirements for stakeholders.	
<b>Entry Criteria</b> (Requirements: Coherent)	
<b>Work Products Associated with the Entry Criteria</b>	
1 List of personification scenarios	
<b>Tasks</b>	
1 Define work teams. 2 Prepare the necessary technical and didactic inputs. 3 Define the date, time, and place of the session. 4 Select and invite stakeholders and key partners. 5 Design formats: attendance, informed consent, use of image and data. 6 Present the work team. 7 Present the context of the session. 8 Present the activities and time structure. 9 Present the available technical and didactic inputs.	10 Organize participants into work teams. 11 Execute the activities with the tools and inputs defined for the session. 12 Present the results obtained by the participants. 13 Foster the discussion of the results and the elaboration of conclusions. 14 Collect results, audiovisual evidence, and informed consent forms. 15 Digitize the information obtained. 16 Synthesize the participants' comments. 17 Identify improvements or relevant contributions. 18 Prepare the conclusions and the session report.
<b>Completion Criteria</b> (Requirements: Coherent)	
<b>Work Products Associated with the Completion Criteria</b>	
1. List of features by scenario	

**Figure 9.** User Persona activity specification card.

For the “User Persona” co-creation technique, the scenarios and associated user types should be available in any chosen physical or digital format. Other elements that facilitate stakeholder participation (markers, notes, paper sheets, and paper tape, or digital tools like tablets, touch screens, and electronic pens) should also be provided.

For the successful execution of this technique, consider the following recommendations:

- All participant groups should analyze all scenarios (one at a time). Workstations with scenarios grouped by context can be set up to allow participant groups to rotate through all associated scenarios.
- Scenarios can be presented in physical or digital format, using the suggested card format or any other format that facilitates participant interaction.
- Participants should write the actions they would like to take and associate them with the analyzed scenario. Cards, sticky notes, or any other physical or digital aids can be used to facilitate this.
- The analysis of results includes collecting and digitizing the functionalities associated with each scenario.

This activity produces the work product “List of Functionalities by Scenario”, which contains the actions that stakeholders would like to take to address the presented problem. This document is crucial as it forms the basis for the functional specification of requirements.

### 5.2.3. Build User Stories

User stories are a standardized format for specifying the functionalities to be included in a software system. The authors in [61] presents a user story format that can serve as a base for developing the stories. In this format, the story content incorporates the keywords “As a user: <user type>”, “I want: <description of what the user wants to do>”, “So that: <justification or benefit the user expects to gain from performing the activity>”. The work



product “List of Functionalities by Scenario”, developed in the previous activity, is used to create the user stories.

This activity results in a set of user stories that describe the functionalities identified with stakeholders during the co-creation session.

5.3. Practice 3—Iterative Development of the Software System

This practice outlines the workflow for developing the software system. It includes the comprehensive specification of requirements prioritized by stakeholders; the definition of the architecture, databases, and design system to be used; and the creation of necessary code components, test management, and the formation of a functional prototype of the software system.

As depicted in the formal representation at the core of Essence (see Figures 10–12), this practice is entirely carried out during the development phase, and the work products are produced by team members with roles such as assistant, manager, stakeholder, analyst, architect, designer, developer, and tester.

In Figure 13, the specification card for this practice is presented. The entry criteria relate to the “requirements” alpha, which must be in the “coherent” state. This means that the requirements consistently describe the fundamental functionalities of the software system.

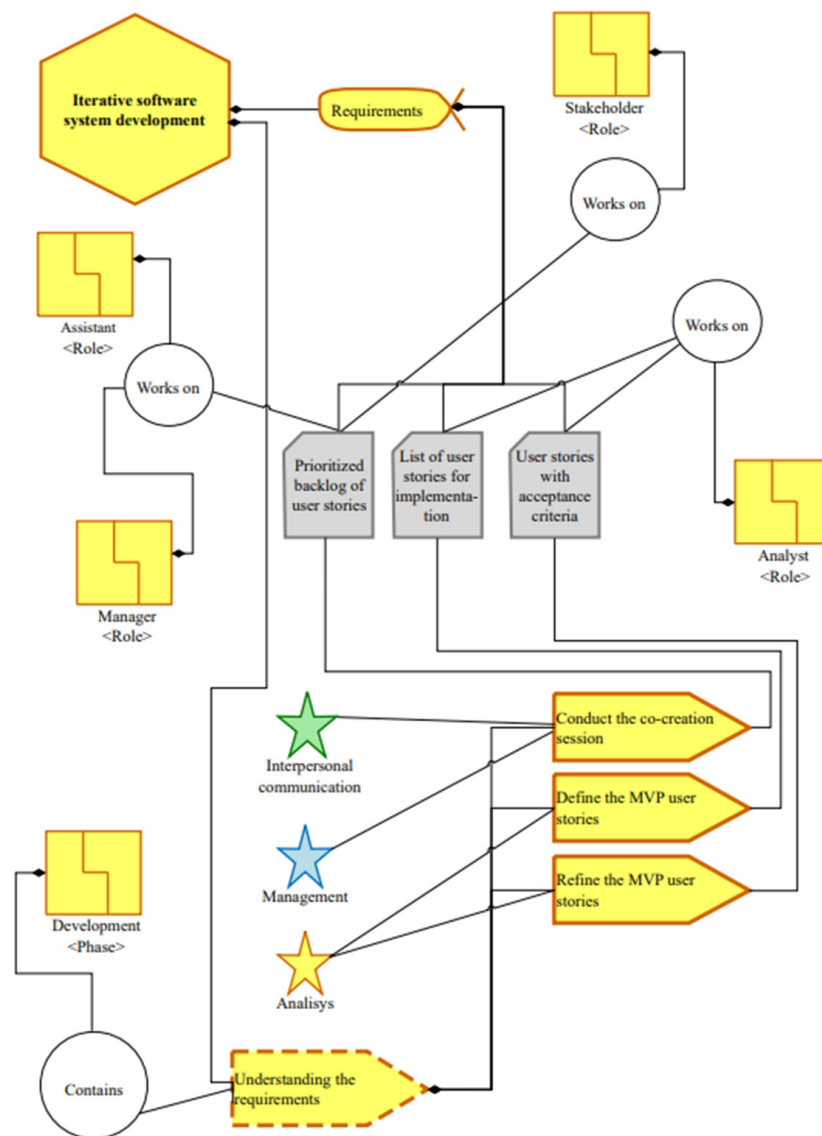
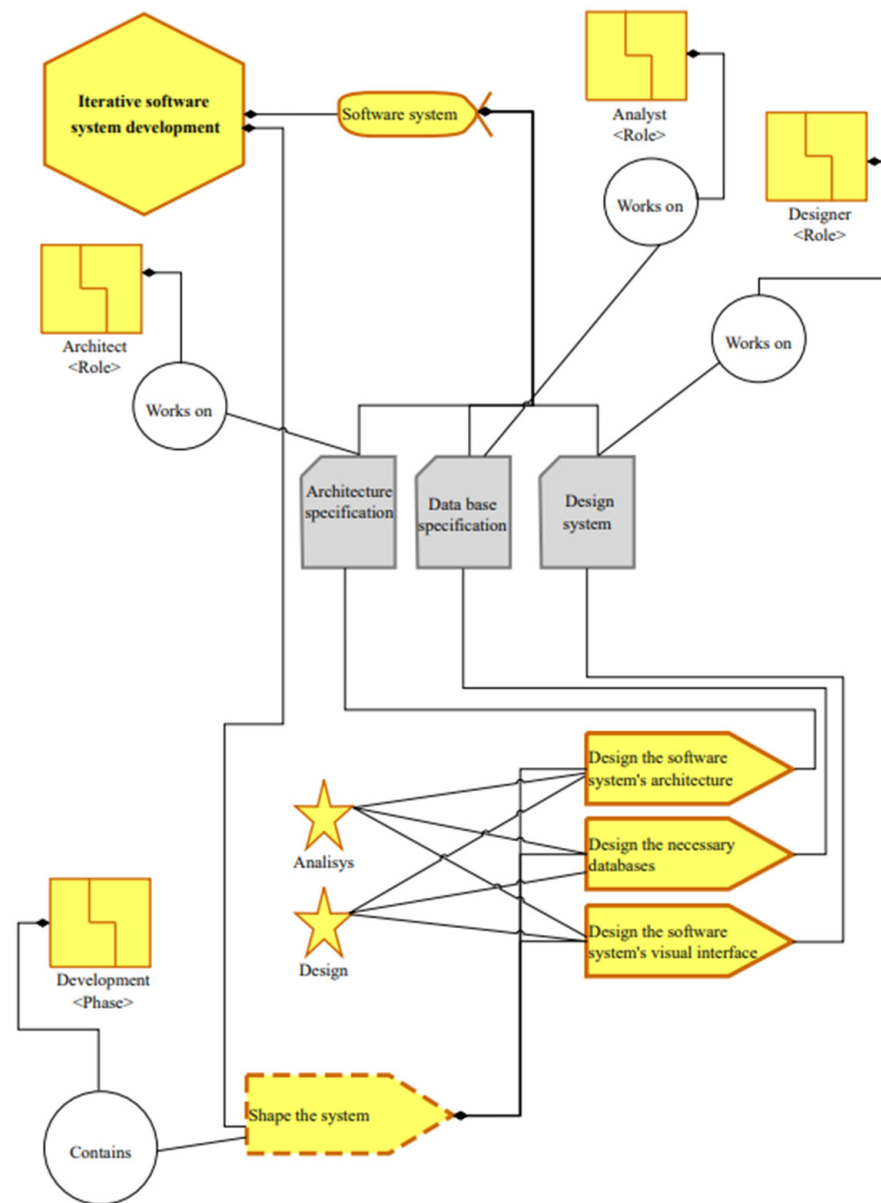


Figure 10. Practice 3—Essence representation 1.



**Figure 11.** Practice 3—Essence representation 2.

The entry criteria are associated with the work products created during the definition phase, specifically in Practice 2: Visual Management of Requirements. Additionally, the work products “List of Key Partners” and “Stakeholder Characterization”, produced during the focus phase, are required.

The exit criteria are associated with the alphas “requirements”, “stakeholders”, and “software system”, which, upon completing this practice, will be in the states of “addressed”, “agreed”, and “usable”, respectively.

In this practice, the necessary work products are created to develop the software system and make it available for stakeholders to interact with it directly.

The activities proposed in this practice are grouped into activity spaces: understanding the requirements, shaping the system, implementing the system, and testing the system. These guide the team in creating work products from the prioritization of requirements to the development of a complete and functional prototype. These activities are described in the proposed order of execution.

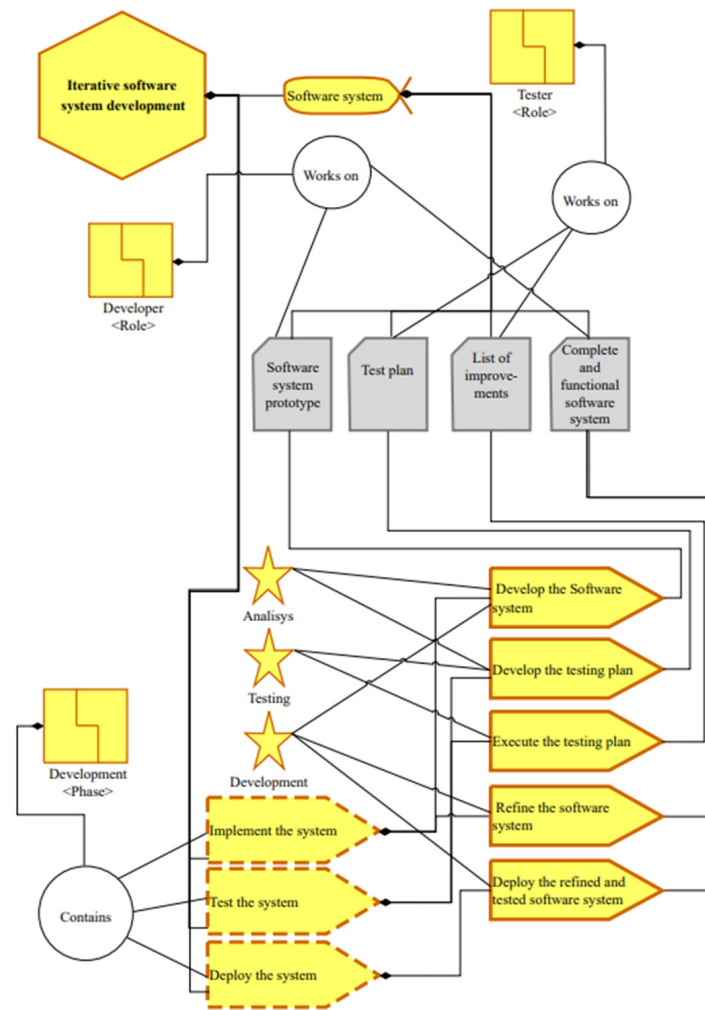


Figure 12. Practice 3—Essence representation 3.


	<b>Name:</b> Iterative Development of the Software System
<b>Description:</b> This practice guides the development of the software system.	
<b>Entry Criteria</b>	
(Requirements: Coherent)	
<b>Work Products Associated with the Entry Criteria</b>	
<ol style="list-style-type: none"> <li>1 List of persona scenarios</li> <li>2 List of functionalities by scenario</li> <li>3 User stories</li> <li>4 List of key partners</li> <li>5 Stakeholder characterization</li> </ol>	
<b>Completion Criteria</b>	
(Requirements: Addressed) (Stakeholders: Agreed) (Software System: Usable)	
<b>Work Products Associated with the Completion Criteria</b>	
<ol style="list-style-type: none"> <li>1 List of prioritized user stories</li> <li>2 List of user stories to be implemented</li> <li>3 User stories with acceptance criteria (complete specification)</li> <li>4 Architecture specification</li> <li>5 Database specification</li> <li>6 Design system</li> <li>7 Software system prototype</li> <li>8 Test plan</li> <li>9 List of improvements</li> <li>10 Complete and functional software system</li> </ol>	


Figure 13. Practice 3—Specification card.

### 5.3.1. Conduct the Co-Creation Session

In this practice, this activity is carried out using the “Feature Buying” co-creation technique. This technique is based on the participation strategy proposed in [9]. This activity involves a metaphorical walk-through different scenarios outlined in the user stories. Workstations are set up where user stories are presented, grouped by user type. Multiple scenarios can be presented per station, if they arise from related problems or the same context.

Different physical or digital-card-based formats can be used to describe the user stories available at each station, facilitating their analysis and stakeholder participation. Once the stories at a station have been analyzed, the participant groups prioritize the stories from most to least relevant based on their judgment. The “100 Dollar” technique is proposed for this purpose [62]. This technique involves assigning each participant 100 points or dollars to distribute among the stories at each station (100 dollars per group of stories).

Figure 14 shows the specification card for this activity. As can be seen, the entry criteria correspond to those of the practice and relate to the work products “List of Persona Scenarios” and “User Stories”. These scenarios and stories are used as the basis for creating the cards or other strategies for presenting the user stories and defining the workstations based on the persona scenarios. Additionally, other elements that facilitate stakeholder participation (markers, sticky notes, and tape, or digital tablets, touch screens, stylus pens, etc., depending on the selected format) are required.

	<b>Practice:</b> Iterative Development of the Software System <b>Activity [1]:</b> Conduct the Co-Creation Session <b>Approach:</b> Iterative <b>Activity Space:</b> Understand Requirements
<b>Description:</b> This activity facilitates the selection of the most relevant requirements for stakeholders.	
<b>Entry Criteria</b>	
(Requirements: Coherent)	
<b>Work Products Associated with the Entry Criteria</b>	
1 List of persona scenarios 2 User stories	
<b>Tareas</b>	
1 Define work teams. 2 Prepare the necessary technical and didactic inputs. 3 Define the date, time, and place of the session. 4 Select and invite stakeholders and key partners. 5 Design formats: attendance, informed consent, use of image and data. 6 Present the work team. 7 Present the context of the session. 8 Present the activities and time structure. 9 Present the available technical and didactic inputs.	10 Organize participants into work teams. 11 Execute the activities with the tools and inputs defined for the session. 12 Present the results obtained by the participants. 13 Foster the discussion of the results and the elaboration of conclusions. 14 Collect results, audiovisual evidence, and informed consent forms. 15 Digitize the information obtained. 16 Synthesize the participants' comments. 17 Identify improvements or relevant contributions. 18 Prepare the conclusions and the session report.
<b>Completion Criteria</b>	
(Requirements: Acceptable)	
<b>Work Products Associated with the Completion Criteria</b>	
1 List of prioritized user stories	

**Figure 14.** Feature Buying activity specification card.

To properly develop this co-creation technique, the following recommendations should be considered:

- All participant groups should analyze and prioritize all available user stories.
- The prioritization format to use should contain the code or identifier of the station and stories, as well as enough space for all participants to assign their desired number of points or dollars. At the end of all sessions, the values assigned to each story are summed to determine the priority order.

This activity produces the “List of Prioritized User Stories” work product. This document contains the user stories grouped by user type and ordered according to the priority established by the participants in the co-creation session. As seen in the specification card Figure 14, the exit criteria indicate that the “requirements” alpha is in the “acceptable” state, implying that the requirements represent a software system acceptable to the stakeholders.

#### 5.3.2. Define a Minimum Viable Product (MVP)

The minimum viable product is a set of functionalities that meet the primary needs of stakeholders. This list of functionalities is defined by the team’s analyst based on the scope of the smart city initiative and the available plant capacity. The MVP also includes complementary functionalities necessary for implementing a software system, such as logging in and out of the system, configurations, and more.

#### 5.3.3. Refine the MVP User Stories

The user stories selected in the previous activity do not contain all the necessary elements to describe the software system’s functional requirements and lack the specificity needed to reach an agreement among the development team members. Therefore, the stories that make up the MVP must be refined through acceptance criteria. Acceptance criteria are conditions that must be met for the user story to be considered complete. A user story can have as many acceptance criteria as needed to define the functionality fully.

This activity’s work product is a requirements specification document that formally describes each user story along with the associated acceptance criteria.

#### 5.3.4. Design the Software System Architecture

Designing the software system architecture is a fundamental activity that involves defining the structure, modules, interfaces, and relationships between components to build a coherent and efficient system. In this activity, the architect makes key decisions about the system’s overall organization; communication between modules; security; performance; and the selection of technologies and platforms for development, deployment, and final use.

The “Architecture Specification” work product is a set of technical documents containing all the system’s characteristics from an internal perspective.

#### 5.3.5. Design the Necessary Databases

This activity involves detailing the characteristics for the structure and management of information in the software system. It is necessary to establish types of databases, specific elements such as tables to be used, fields, primary and foreign keys, data types, and integrity rules. This specification ensures that the database meets the requirements, aligns with the system architecture, is efficient in data storage and retrieval, and addresses security considerations for data protection.

#### 5.3.6. Design the Software System’s Visual Interface

This activity involves defining aspects such as the visual layout of elements, graphic style, navigation, and user experience (UX). These elements represent the interface’s appearance and behavior and consider usability, accessibility, and visual consistency. The “Design System” work product includes the software system’s graphical interfaces, logos, icons, and other elements required for system development. Given the dynamic context of the method, this activity involves studying the stakeholders to properly define the visual interface style. Therefore, the design system includes a user research report.

### 5.3.7. Develop the Software System

This activity involves creating the necessary code files to shape the software system and turn it into a usable product that meets the established requirements. This activity follows the guidelines defined in previous activities regarding the software system's architecture and design. Best practices for coding, version management, and issue control should be considered to avoid conflicts and errors in the created files. Finally, this activity produces the "Software System Prototype" work product, which is a version of the system that includes all the functionalities defined in the MVP but has not been tested and is subject to improvements.

### 5.3.8. Develop the Test Plan

This activity involves defining and documenting a detailed strategy to evaluate various aspects of the software system, such as functionality, performance, and security. The test plan addresses key issues such as the functionalities to be tested, how the tests will be conducted, who will execute them, when they will be performed, how results will be reported, what metrics will be used, and more. This document is essential for ensuring software quality, identifying errors, and providing a detailed guide for adequately testing the system.

### 5.3.9. Execute the Test Plan

Executing the test plan involves carrying out the tests defined in the plan developed in the previous activity. Errors identified during the execution of the test plan are documented to obtain performance statistics and improvement actions to refine the software system. This activity produces the "Improvement List" document, which describes all the details of the identified errors and additional observations that facilitate the identification of the code files that need modification.

### 5.3.10. Refine the Software System

This activity involves correcting the errors identified during the test plan execution by modifying code files, database structures, communication services, and more. The "Complete and Functional Software System" work product is an executable that can be used to showcase the results to stakeholders, allowing them to interact with a complete version of the system.

## 5.4. Practice 4: Experimental Validation of Stakeholder Satisfaction

This practice outlines the approach to validate the software system through user experiences, together with the stakeholders. As illustrated in the formal representation at the core of Essence (see Figure 15), this practice is entirely conducted during the validation phase, and the work products are produced by team members with roles such as manager, assistant, stakeholder, tester, and designer.

In Figure 16, the specification for this practice is presented. The entry criteria are associated with the alphas of requirements, stakeholders, and the software system, which must be in the states of addressed, agreed, and usable, respectively. This implies that stakeholders agree that the most important requirements have been addressed, and a usable software system has been created.

As seen in the specification card for the practice, the entry criteria are linked to some work products created in previous practices. On the other hand, the completion criteria are associated with the alphas of the software system, stakeholders, opportunity, and requirements, which at the end of this practice, are in the states of ready, satisfied for deployment, addressed, and fulfilled, respectively.

The proposed activities in this practice are grouped in the activity space to ensure stakeholder satisfaction. Therefore, the work products created a focus on obtaining opportunities for improvement and direct feedback from stakeholders based on their interaction with the system. These activities are described in the proposed execution order below.

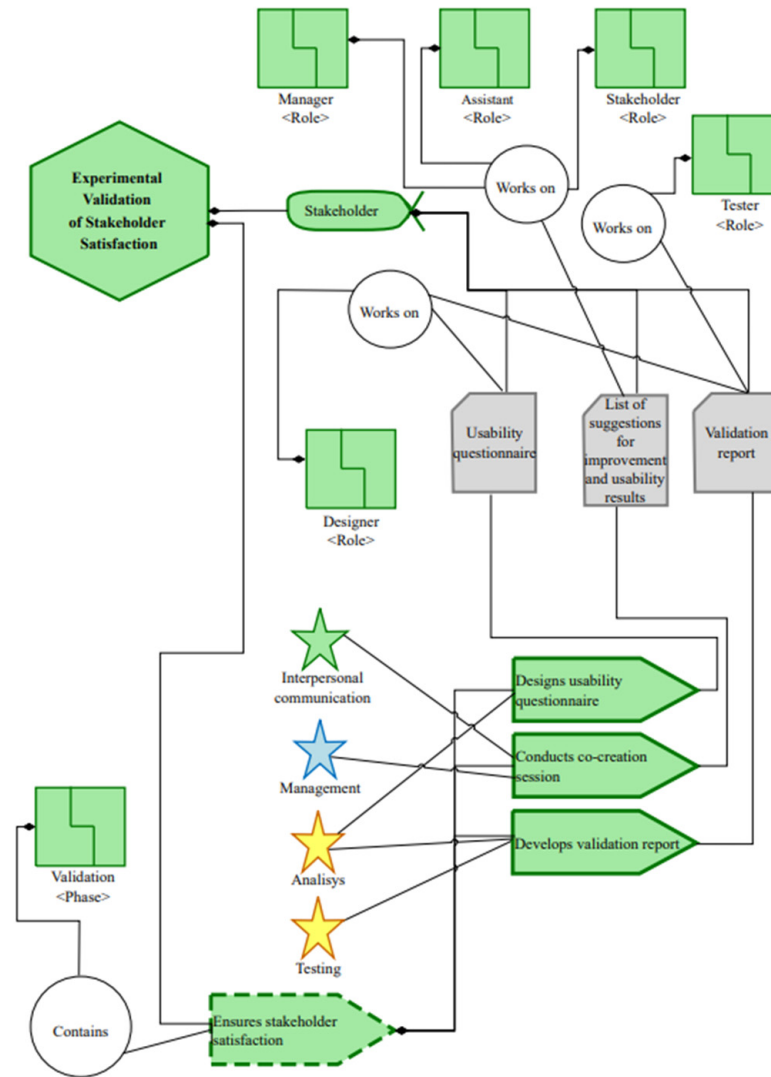


Figure 15. Practice 4—Essence representation.

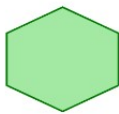
	<b>Name:</b> Experimental Validation of Stakeholder Satisfaction
<b>Description:</b> This practice guides the validation of stakeholder satisfaction through the use of the developed software system.	
<b>Entry Criteria</b>	
(Requirements: Addressed) (Stakeholders: Agreed) (Software System: Usable)	
<b>Work Products Associated with the Entry Criteria</b>	
1 Complete and functional software system	
2 List of key partners	
3 Stakeholder characterization	
4 User stories with acceptance criteria (complete specification)	
<b>Completion Criteria</b>	
(Software System: Ready) (Stakeholders: Satisfied for Deployment) (Opportunity: Addressed) (Requirements: Met)	
<b>Work Products Associated with the Completion Criteria</b>	
1 Usability questionnaire	
2 List of improvement suggestions	
3 Validation report	

Figure 16. Practice 4—Specification card.

### 5.4.1. Design the Usability Questionnaire

This activity involves creating a structured set of questions aimed at gathering specific data about users’ experiences with the developed system. The questions focus on aspects such as navigation, information clarity, and overall user satisfaction. The goal is to develop a clear questionnaire that provides meaningful information to improve the system’s usability. To carry out this activity, the objectives, target audience, and desired metrics must be considered to create an effective tool that collects both quantitative and qualitative data to identify improvements in design and user experience.

### 5.4.2. Conduct the Co-Creation Session

In this practice, this activity is conducted using the “Experimentation Scenarios” co-creation technique. In this activity, stakeholders can interact with the developed prototype and validate the inclusion of functionalities, visual aspects, and performance, among other aspects of the software system.

To conduct the co-creation session with the mentioned technique, inputs created in previous practices are used. During the session, participants analyze the user stories representing the functionalities included in the software system prototype and interact with it to validate its operation. This allows participants to see firsthand the visual aspects of the system, or the steps required to perform a specific action, and to write down improvement suggestions they deem necessary.

The user stories are presented in the format described earlier and are organized so that all participants validate all aspects of the system in a logical order, meaning the actions to be performed must be consecutive. At the end of the session, participants complete the usability questionnaire and provide additional recommendations if they see fit.

Figure 17 shows the specification card for this activity. As can be seen, the entry and exit criteria remain unchanged and correspond to the entry criteria of the practice, as the state of the alphas does not change during the co-creation activity.


	<b>Practice:</b> Experimental validation of stakeholder satisfaction <b>Activity [2]:</b> Conduct the co-creation session <b>Approach:</b> Experimental <b>Activity space:</b> Ensure stakeholder satisfaction
<b>Description:</b> This activity facilitates stakeholder interaction with the developed software system, which in turn allows for identifying improvement actions and understanding opinions about the most relevant aspects of it.	
<b>Entry criteria</b>	
(Requirements: Addressed) (Stakeholders: In agreement) (Software system: Usable)	
<b>Work products associated with the entry criteria</b>	
1 Complete and functional software system 2 List of key partners 3 Stakeholder characterization 4 User stories with acceptance criteria (complete specification) 5 Usability questionnaire	
<b>Tasks</b>	
1 Define the work teams. 2 Prepare the necessary technical and educational inputs. 3 Define the date, time, and place of the session. 4 Select and invite stakeholders and key partners. 5 Design formats: attendance, informed consent, image and data usage. 6 Present the work team. 7 Present the context of the session. 8 Present the activities and the time structure. 9 Present the available technical and educational inputs.	10 Organize participants into work teams. 11 Execute the activities with the defined tools and inputs for the session. 12 Present the results obtained by the participants. 13 Encourage discussion of the results and the formulation of conclusions. 14 Collect results, audiovisual evidence, and informed consent forms. 15 Digitize the obtained information. 16 Synthesize participant comments. 17 Identify relevant improvements or contributions. 18 Prepare the conclusions and report of the session.
<b>Completion criteria</b>	
(Requirements: Addressed) (Stakeholders: In agreement) (Software system: Usable)	
<b>Work products associated with the completion criteria</b>	
1 List of improvement suggestions and usability results	

Figure 17. Experimentation scenario activity specification card.



To properly conduct this co-creation technique, the following recommendations should be considered:

- User stories should be grouped by user type, ensuring that the actions to be executed are continuous and in the correct order, for example: a user must create an account in the application before they can log in.
- All participants should have access to the prototype; therefore, it is necessary to ensure that there are enough mobile or fixed devices to conduct the activity. If the selected platform is mobile, participants' devices can be used if they are compatible and there is sufficient connectivity to install and run the prototype.
- Participants' contributions can be collected using sticky notes, index cards, tablets, or any other means that allows participants to write down their improvement suggestions.
- Usability questionnaires are handed out at the end of the session so that participants have already interacted with the prototype being evaluated.

This activity produces the work product "List of Improvement Suggestions and Usability Results", which contains the improvement suggestions grouped by user story and the quantitative and qualitative results of the usability study. This document serves as the necessary input for the subsequent creation of the validation report.

#### 5.4.3. Create the Validation Report

This activity involves documenting the results and conclusions derived from the software system validation process. This report includes usability test results; improvement suggestions collected during the co-creation session; and other details about requirement fulfillment, such as identified issues and possible corrections. This document provides a basis for making decisions about deploying the software system, as well as offering recommendations for future improvements.

Based on the validation report, the development team determines whether it is appropriate to re-execute the last four activities of Practice 3 to improve the system or continue with the deployment process using the current version. Although all stakeholder suggestions should be analyzed, it is the development team's discretion to adopt, discard, or propose recommendations for a future version of the system.

## 6. Validation

The method for developing software systems in the context of citizen-centered smart cities is validated through consultation with experts in software engineering and smart cities. The focus group method is implemented to gather improvement actions that allow for the adjustment of the results and achieve an acceptable level of consensus, indicating a high degree of completeness in the proposed solution [63–65].

A focus group is a widely used expert consultation method in various areas of engineering and science to evaluate the quality of research results. This method allows for the collection of both quantitative and qualitative data that describe individual opinions and the group consensus level of a panel of experts on the proposed topic [35].

As described in methodology section, the objectives of the validation process are initially defined, and the necessary materials for carrying out the process are prepared. In this research, the following objectives were defined:

Objective 1: Validate the citizen participation mechanisms included in the method as an alternative to facilitate the inclusion of stakeholders and other actors in the territory in the process of developing software systems in the context of citizen-centered smart cities.

Objective 2: Validate the activities and work products incorporated into the method for developing software systems in the context of citizen-centered smart cities.

Objective 3: Validate that the proposed method facilitates the development of software systems in the context of citizen-centered smart cities.

According to the process described by [35], a focus group session is characterized by the interaction between the experts and the subject of study, as well as the researcher. During a typical session, the researcher provides a concise presentation of the subject of

study, which is what is being put forward for expert consideration. The researcher may propose a set of motivating questions to promote expert participation in an open discussion about the key elements of the subject of study. Likewise, experts may ask questions to the researcher to clarify their doubts. Finally, the researcher provides the experts with a tool to collect their opinions, usually a questionnaire with open-ended and scale-based questions, to gather both qualitative and quantitative results.

To validate the results of this research, a questionnaire was designed with eight items featuring open-ended questions and a five-point Likert scale. The items included in the questionnaire are as follows:

Item 1: The proposed activities in the focus phase facilitate the identification of relevant stakeholders and key partners, as well as the preparation of tools, schedules, or other elements necessary for carrying out the subsequent activities.

Item 2: The “Anchored Sailboat” co-creation technique facilitates the characterization of contexts and the identification of needs, barriers, or impediments in various critical areas of a smart city.

Item 3: The “User Persona” co-creation technique facilitates the specification of valuable ICT products and services for stakeholders by identifying functionalities and roles within a specific context of a smart city.

Item 4: The “Feature Buying” co-creation technique facilitates the selection of the most relevant functionalities to form the minimum viable product to be developed.

Item 5: The proposed activities in the development phase facilitate the design and development of software products based on the selected functionalities.

Item 6: The “Experimentation Scenarios” co-creation technique facilitates stakeholder interaction with the developed software product and the identification of improvements or additional functionalities.

Item 7: The co-creation techniques proposed in the method are appropriate mechanisms to facilitate stakeholder participation and integration in the software design and development process within the context of smart cities.

Item 8: The proposed method facilitates the development of software systems in collaboration with stakeholders in the context of citizen-centered smart cities.

Additionally, support materials were prepared for the presentation, namely, forms to collect specific data from the experts, and to inform them about the use of the obtained results, image management, and personal data handling.

The expert profile was defined to include professionals with education and experience in relevant areas of knowledge, in this case, smart cities and user-oriented software system development, using stakeholder participation strategies and involving other parties as much as possible. To ensure this profile, professionals with postgraduate degrees (master’s and preferably doctoral degrees) in areas related to computer science and experience in public or private projects focused on developing ICT products and services with a strong social emphasis were consulted. The following are the profiles of the professionals who agreed to participate as experts in the validation of the main product of this research:

Raúl Mazo: PhD in Computer Science from Pantheon-Sorbonne University, where he worked as an associate professor from 2012 to 2019. Since 2019, he has been a full professor at ENSTA Bretagne (in Brest, France). Since 2011, he has led the Variamos research program and team, involving academics, industry professionals, and students from various countries. With the software associated with this research program ([variarnos.com](http://variarnos.com)), Raúl enriches his courses in Software Engineering and has participated in more than 10 national and European research projects. His research work, which he regularly presents at international journals and conferences (over 150 to date), has been distinguished with six best paper awards. Raúl has advised several companies in France, Belgium, and Colombia on issues related to reuse and variability management. Before working in academia, Raúl worked for several years as a software developer, analyst, and telecommunications engineer for companies in Colombia and France.

Roberto Carlos Hincapié: electronic engineer, and Master and Doctor of Engineering. He has been the dean of engineering at Pontifical Bolivarian University for six and a half years, as well as a faculty director and senior researcher at MinCiencias and the GIDATI research group. His research focuses on modeling, optimization, and system simulation. Additionally, he has participated in technology projects with a social focus in areas such as health and public services.

Luis Fernando Londoño: Systems engineer from the University of Antioquia (1990), specialist in management for engineers from Pontifical Bolivarian University (1999), and Master’s in Engineering from EAFIT University (2019). He has 35 years of experience in the software industry, having held positions in programming, architecture, project management, and technological innovation, as well as being a co-founder of tech-based companies since 1994. Currently, he is the co-founder and leader of Innovation and Technological Development at Koral Advanced Technology, a company that develops software-intensive solutions for the agricultural and mining sectors in the country, as well as data analytics solutions. He is part of the ecosystem surrounding the GIDIA artificial intelligence group at the National University and conducts research on modeling languages for dynamic variability systems with research groups from various universities. Additionally, he has been affiliated with academia as a professor at the University of Antioquia, EAFIT University, and the University of Quindío, teaching courses related to software engineering, particularly in areas such as modeling languages, methods, and frameworks for software development.

The session was conducted synchronously via videoconference due to the geographical distance of the experts. During a two-hour work session, all the proposed activities were carried out, and improvement actions and quantitative data were collected, allowing for consensus levels to be defined and the proposed method to be refined. Table 2 presents the improvement actions that emerged from the focus group session and the activities carried out to address these needs. The improvement actions are divided into two groups: elements that are present in the subject of study (method) but need to be described in greater detail to facilitate understanding, and elements that are not present in the subject of study and must be added to ensure the proposed objectives are met.

**Table 2.** Focus group improvement actions.

Suggestion	Improvement Action
Group 1	
It is important to have a broader vision regarding the roles involved in the method.	A specific section is added to describe the roles and their responsibilities.
Co-creation techniques should be documented in a more structured way to facilitate their understanding and how they complement the method.	A unified structure is adopted for the description of co-creation techniques, and additional recommendations are added for their correct implementation.
Include in the state-of-the-art references to other co-creation techniques that can be used to broaden the range of available techniques.	Given that the proposed techniques have been adapted to meet the needs of the context, it is difficult to find other techniques that can be used without modifications. However, the state of the art and throughout the document suggest references to various techniques for value generation.
Support the proposal with references to existing methodologies and frameworks.	Similar methodologies and other proposals are detailed in the background section (not included in the documentation sent to the experts).
Make clear the type of software to which the method is oriented, since there are many types and each requires certain processes, for example, critical systems, such as commercial aircraft, require methods like the V-model to obtain the necessary certifications and be able to commercialize them.	It is made clear in the description of the method that the focus is on the development of citizen-oriented software systems during their daily lives and that can be executed on personal computers or mobile devices

Table 2. Cont.

Suggestion	Improvement Action
Group 2	
Use a recognized process modeling language or notation in academic and industry environments to represent the processes proposed as part of the method.	Employ the core of the Essence of software engineering as a standard for defining the practices that make up the method, its activities, and its corresponding representation.
Unify the terms related to software engineering, since common terms are used, but may not be interpreted as synonyms by some readers.	Adjust terms according to the Essence Kernel glossary.
Modify the conceptual scheme of the method so that it represents an iterative and incremental process.	Develop a new conceptual scheme of a Cartesian plane type.
Describe in more detail the development phase, the proposed activities, the work products that are produced, and how to work with user stories.	All practices are represented in the Essence core, and the activities, roles, and work products that are produced are described.
Add criteria to user stories, since to work adequately, acceptance criteria must be defined to formalize the functional requirement.	The work products produced in the development phase are refined to detail the functional and technical specification documents.

Upon reviewing the results, it was evident that the experts considered it important to make some adjustments to how the elements comprising the method were described. Regarding the co-creation techniques, they emphasized the importance of providing more detailed descriptions of the work products generated by each technique, the inputs required, and the tools that can be used to implement a particular co-creation technique.

Similarly, they identified deficiencies in the descriptions of technical work products and highlighted the need for more detailed specifications on how stakeholders and other territorial actors participate in identifying, specifying, and prioritizing the requirements that describe functional needs.

As previously mentioned, the quantitative results were captured using Likert scale questions with five options, where a value of 1 represents “completely disagree”, and a value of 5 represents “completely agree”. Based on this scale, a set of ranges was established to associate the results with decimal values, thus defining a consensus level by item, by expert, and an overall level considering all the results. The scale levels are presented in Table 3, and the total quantitative results can be seen in Table 4.

Table 3. Scale levels.

Level	Assignment	Range
Strongly Agree	5	4.2 < N ≤ 5
Agree	4	3.4 < N ≤ 4.2
Neutral	3	2.6 < N ≤ 3.4
Disagree	2	1.8 < N ≤ 2.6
Strongly Disagree	1	1.0 < N ≤ 1.8

The experts’ ratings fall between “Agree” and “Strongly Agree”, indicating a high level of acceptance for the co-creation techniques proposed in the method, the additional activities, and the overall structure of the method for software systems development in the context of citizen-centered smart cities. Similarly, Table 5 presents some positive comments from the experts that complement the improvement actions and highlight the favorable aspects that, according to the experts’ knowledge and experiences, are key characteristics to emphasize in the proposed method.

Finally, the validation process of the method not only confirmed its relevance and acceptance among experts but also identified key areas that have already been improved and strengthened. The experts positively assessed the co-creation techniques and the

overall structure of the method, indicating that it is a robust and effective tool for the development of software systems within the context of citizen-centered smart cities.

**Table 4.** Focus group quantitative results.

Item	Exp 1	Exp 2	Exp 3	Item Average	Likert Scale
1	4	5	4	4.3	Strongly Agree
2	4	5	5	4.7	Strongly Agree
3	4	5	4	4.3	Strongly Agree
4	4	5	5	4.7	Strongly Agree
5	3	5	4	4.0	Agree
6	4	4	4	4.0	Agree
7	4	5	5	4.7	Strongly Agree
8	3	5	5	4.3	Strongly Agree
Expert average Likert level	3.8	4.9	4.5	4.4	Strongly Agree
	Agree	Strongly Agree	Strongly Agree	Strongly Agree	

**Table 5.** Focus group positive comments.

Comment
The proposed process is indeed very comprehensive and, particularly in a multi-user scenario, effectively facilitates the collection of expectations, requirements, and perspectives for platform development. I am not aware of other similar processes that so comprehensively address the challenges of development in this context.
The “Feature Purchasing” technique is particularly strong for the phase in which it is applied.
It is a proposal of great value for the academic community and opens possibilities for new research and even industrial applications.
It is an original proposal that aims to contribute to existing methods for developing intensive software solutions.

The observations made were considered, deepening the description of certain elements and adding new aspects that were not initially considered. This has made the method more understandable and effective, facilitating its application in real-world scenarios. The positive comments and improvement suggestions provided by the experts have been integrated, solidifying the method as a comprehensive solution tailored to the complexities of citizen participation in smart cities.

### 7. Use Case

The described method was tested in the development of an initiative to identify and propose software-based solutions to address the challenges faced by citizens in Medellín, Colombia, regarding public safety. This section presents the results obtained from the execution of the activities and techniques suggested in the four practices that make up the method, along with the characteristics of the resulting software system prototype.

The initiative was framed within the objectives of the research and development project that originated this article, as mentioned in the acknowledgments section. The justification of this project includes the necessary inputs to use the method, such as statistics and data on the critical area of interest (public safety), contact information for potential key partners, and a list of the most affected sectors based on previous studies, among others. Likewise, the project’s nature facilitated the formation of work teams with the required profiles, mostly available in the list of affiliated professionals or with some working relationship. It is worth mentioning that the authors of this article were part of this initiative in management and analysis roles.

Based on the available inputs, the activities described in practice 1 were carried out, resulting in the following work products:

List of key partners: Key partners were identified, including public and private companies related to public safety and other public entities involved in promoting social initiatives. Noteworthy are companies focused on training and certification for security personnel, private security companies (personal protection and general security in companies or public places), public security companies, Empresa para la Seguridad Urbana (ESU), the Ruta-N

Business Center, and the Valle del Software Center. Additionally, experts in public safety were identified, such as professors, retired police members working as security consultants, and analysts with an emphasis on society and civic culture, among others.

Stakeholder characterization: In the critical area of public safety, stakeholders were selected, including residents of commune La Candelaria in Medellín, some public and private security companies based or operating in the commune, educational institutions (members and security personnel), and community organizations.

List of relevant contexts in interest area: The contexts in which the most problems were identified were selected based on data and statistical analysis and consultation with public entities and experts in public safety. The contexts identified included urban safety, digital security, and home security, which were used to carry out the co-creation session suggested in practice 1.

List of problems by context and affected users: In the co-creation session, 25 participants, including community members and delegates from public and private companies and institutions, identified many problems in the context of urban safety. Table 6 lists some of the most relevant problems in this context.

**Table 6.** Urban security problems.

Context	Problem
Urban security	Robbery in commercial establishments
	Vehicle and auto parts theft
	Robbery in public transportation
	Homicides in public areas
	Presence of criminal gangs
	Conflicts with neighbors
	Gender-based violence
	Armed robbery
Clashes among armed groups	

Once the most relevant context and associated problems were identified, practice 2 was implemented, resulting in the following work products:

List of scenario personas: The proposed personas align with the problems identified by participants in the practice 1 co-creation session. Everyday scenarios involving “common” criminal acts were prioritized, such as armed robbery on public transport, noise disputes between neighbors, and car part thefts in public areas, among others. Additionally, two user types were identified from the work products of practice 1 and the defined scenarios: a user who witnesses or suspects a crime and a user who is a direct victim of a crime.

List of features: The co-creation session had 22 participants who contributed their experiences and knowledge to define a set of features, presented in Table 7.

**Table 7.** Urban security features.

Context	Feature
Urban security	Report crimes I witness or fall victim to through a simple and fast information system.
	View crimes reported by me or other citizens in areas of my interest.
	Contribute to security events reported near my workplace.
	Attach evidence like photos, audio, or videos to the security reports.
	Generate silent panic reports.
	Report a security event anonymously.
	Notify family and friends about the security events I report.

User stories: Based on the defined features, a set of user stories was created in the suggested format, describing the scenarios and their associated characteristics. This resulted in user stories associated with the two identified profiles. Here is an example:

As a: citizen who witnesses or suspects a crime.

I want: to quickly report the crime I am witnessing or suspecting and attach audiovisual evidence.

So that: I can send an alert to the relevant authorities in a timely manner and facilitate incident response.

After defining the user stories, the activities of practice 3 were carried out, resulting in the following work products:

Backlog of prioritized user stories: The practice 3 co-creation session included 35 participants, such as business owners, community members, security company personnel, and educational institution members. Using the suggested technique, the user stories deemed by participants to provide the most value to the system were selected.

List of user stories for implementation: The prioritized stories were complemented with additional required features to shape the system, including functionalities for system access and logout, general reconfigurations, and visual customization. Given the nature of the desired features, a mobile software system was identified during the analysis to ensure the solution was accessible anytime.

User stories with acceptance criteria: The stories selected for implementation were supplemented with acceptance criteria and low-fidelity mockups, creating a complete set of functional requirements used as a starting point for the more technical activities of practice 3.

Architecture specification: Due to the scope of the research and development project, the technical decision was made to develop a mobile solution with backend logic based on microservices, enabling independent service reuse, scalability, and updating. The microservices were built in Java using the Spring Boot framework and employed technologies such as Docker and Apache Kafka, commonly used in enterprise environments for their extensive documentation and long-term support. The Flutter framework was chosen for front-end development due to its usability and versatility in generating executables for iOS and Android devices. Development, testing, and controlled deployment were carried out on project servers (GNU Linux-based machines with high processing capacities). Additionally, a GitLab server was installed and configured to ensure controlled access and code file backups through automatic backups.

Database specification: Two database engines, PostgreSQL and MongoDB, were selected—one relational database to manage users and store typed data and a non-relational database to store emergency event information and related evidence in Base64 format. Both databases were separately configured using Docker containers with security and access control best practices.

Design system: Screens or views were created using the Figma tool, with prior studies on the target community informing the style, color scheme, and other usability elements, designing a user-friendly and easy-to-use solution (see Figure 18).

Software system prototype: Development was carried out in twelve (12) short weekly cycles covering all the previously defined backlog functionalities. Three versions were defined, limited by the included functionalities, in the following order:

Version 1: Registration, login and logout, data updates, and access to the system's main menu.

Version 2: Event reporting, evidence upload and modification, and immediate emergency event reporting.

Version 3: Event display on a map.

An internal testing plan was defined and executed for each version to identify improvement actions related to acceptance criteria compliance, such as system logic errors, unexpected error messages, incorrect texts, and form validation. This approach is just one of many documented strategies for managing development. Each work team can define a different approach to develop the functionality set, as long as the expected work products are delivered.

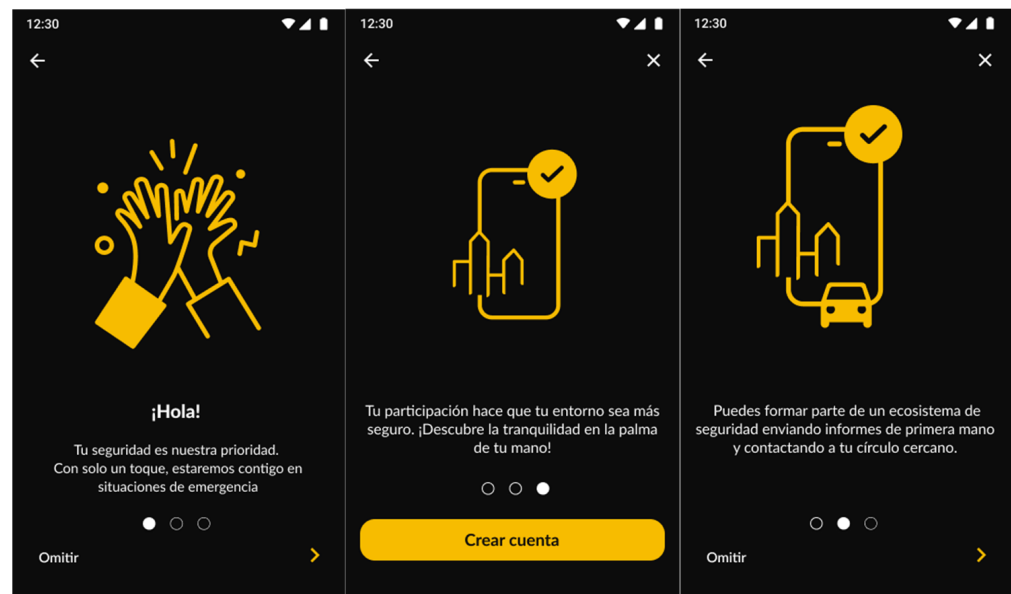


Figure 18. Design system.

Upon completing the release of the versions, a complete and functional software system was obtained that met the defined acceptance criteria and could be used as an input for practice 4, resulting in the following work products:

**Usability questionnaire:** The usability questionnaire design was based on the standardized Computer System Usability Questionnaire (CSUQ) and the System Usability Scale (SUS). This resulted in a 16-item questionnaire in Spanish with Likert scale responses ranging from 1 (strongly disagree) to 5 (strongly agree). The results from this questionnaire were analyzed using the general usability scale to determine the perceived acceptance range by users.

**List of improvement actions and validation results:** At the end of the practice 4 co-creation session, a set of improvement actions were identified to align the solution with stakeholder expectations. This session included 30 participants of diverse demographics, including men, women, young adults, and seniors, with varying levels of experience with mobile devices, generating relevant feedback on system usability, the required steps for actions, requested information, clarity of error messages, amount of available information, and more.

All feedback received and the usability questionnaire results were documented in a validation report delivered to the development team for the necessary corrections. Consequently, development and internal testing activities were re-executed to incorporate the identified improvements. The final system version was presented to a larger audience (100 people) during a subsequent project activity aimed at disseminating the obtained results and as part of a package of ICT products and services developed to address public safety issues from various perspectives. Most feedback was positive regarding the included functionalities, ease of use, and overall system performance.

## 8. Developed System

A mobile application was developed initially for devices based on the Android operating system, allowing community members to report security events in two modes:

**Security events witnessed or suspected:** This type of event is characterized by the user not being directly affected by the criminal act and having sufficient time to capture audiovisual evidence to support the report. The app includes a list of common events (based on prior statistics), such as various types of theft, traffic accidents, domestic violence, and animal issues, among others. Users can quickly select the type of event they wish to report, provide a description via text or voice note, and attach images and videos through



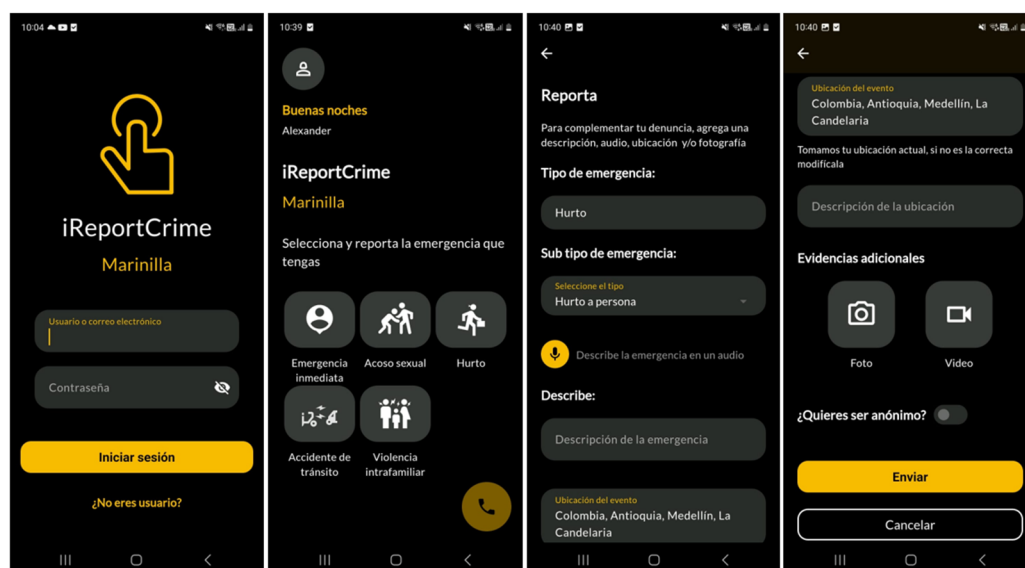
a user-friendly and intuitive camera interface. Additionally, users can configure an event location and describe the location if it differs from the device's current location.

**Immediate panic events:** In these events, the user is directly affected by a criminal act and lacks the time to capture evidence or complete the required form. In such cases, users can trigger an immediate panic event using their device's volume button, creating an "immediate emergency" report with the user's data, date, time, and device location.

Both modes generate notifications for the user's registered emergency contacts and log events, along with all evidence, on a web-based incident management platform accessible to public or private organization members responsible for the area where the event occurred. The specifics of this platform, however, are outside the scope of this article's case study.

Additionally, the app allows users to view events reported by others in the territory on an Open Street Map interface, showing only basic event details—event type, location, and basic description—to protect audiovisual evidence and the reporting user's information.

The application also includes complementary functionalities, such as registering new users; logging in and out; configuring a user profile (updating or modifying information); registering, modifying, or deleting emergency contacts (up to five per user); changing the app color scheme; and activating the immediate emergency listening system (which maps the volume-down button to log an "immediate emergency" event). Figure 19 shows some screenshots taken from a real device.



**Figure 19.** Developed system. Note: Due to the context in which this initiative was developed and the objectives of the funding project, it is not possible to modify the app's language. Therefore, the app interface is shown in Spanish. We apologize for any inconvenience this may cause.

## 9. Identified Limitations

The case study highlighted some limitations in using the method in real-world scenarios:

Prior knowledge of co-creation session development is required: These activities require detailed step-by-step planning, covering everything from preparing required materials (invitations, forms, ID cards, etc.) to collecting and digitizing results. While teams are free to select the strategy that best suits them, it is advisable to define a document base to guide all co-creation-related activities. In this initiative, we relied on our prior work describing a set of activities for conducting co-creation sessions. See the conference paper [66].

Implementing the proposed method requires considerable time, particularly due to the effort involved in co-creation sessions: Each session involves several steps, including attendee registration, material preparation, activity execution, results collection, and final product creation. In this initiative, each co-creation session took approximately two weeks,

totaling eight weeks to complete all sessions suggested in the method. This is an important consideration for work teams, as it distances the method from an agile context.

Building a citizen-centered solution in smart cities that meets all stakeholders' needs depends on the level of participation from these stakeholders: Without successfully engaging members from various interest groups in collaborative work sessions (co-creation sessions), there is a high risk of developing a deficient system. To mitigate this risk, a broad promotion of the initiative is recommended, along with outreach to community leaders and social organizations (e.g., community mothers, sewing workshops, community councils) in order to communicate expectations and emphasize the importance of community involvement, thereby increasing the likelihood of engaging a broad range of participants.

Scope and validation: The validation was limited to a single case study and focus groups involving a specific set of stakeholders. This may restrict the generalizability of the findings to other contexts or smart city areas. Additional validations across diverse environments and cultural settings are required to confirm the method's broader applicability.

## 10. Conclusions

This study introduced a method for developing software systems within citizen-centric smart cities. The method encompasses practices involving citizen participation mechanisms and complementary activities to identify needs, define and prioritize functionalities, develop the systems, and validate the resulting prototypes.

The method incorporates co-creation techniques to engage citizens and other local stakeholders. These techniques aid in producing the necessary outputs for developing software systems in citizen-focused smart cities. By actively involving citizens, the likelihood of creating software systems that closely match their real needs increases.

The method was validated through focus groups, where expert interactions provided valuable qualitative insights and collected quantitative data, establishing a consensus on the method's appropriateness for software development in citizen-centric smart cities. Feedback from experts allowed for adjustments and improvements, resulting in a more comprehensive version that includes essential elements for guiding software development in the analyzed context.

Additionally, an initiative was carried out in which a mobile application focused on mitigating public safety issues was developed. Some aspects that must be considered when using the method in real smart city contexts were identified. Noteworthy are the time required for system development, the need for prior knowledge in collaborative work sessions with multiple stakeholders, and the importance of citizen participation to increase the likelihood of developing systems that meet their needs.

Definitions from the Essence framework were key throughout this process, facilitating the creation of four practices with clearly defined activities and work products. Representation schemes for each practice and created specification cards simplify the method's application and progress analysis.

This software development method for smart cities is a significant contribution, given existing alternatives. As previously described, software systems for smart cities are typically developed using combinations of standardized methods, indicating that development teams lack specific alternatives for addressing challenges in this context.

Additionally, no similar proposals were found to compare the results from the perspective of the practices comprising the method. Most approaches focus on specific smart city areas and lack detailed and standardized activity descriptions or relevant work product development activities. They often only suggest identifying problems and validating developed solutions, leaving many elements unaddressed.

## 11. Future Work

Although the method demonstrated in this study proved useful in a real-world scenario, the study's limitations highlight opportunities for future research and development. Expanding the method's application to diverse smart city contexts, improving the efficiency

of co-creation sessions through digital tools, incorporating additional techniques that could be more easily applied to other critical areas of a smart city, and validating the approach with different stakeholder groups are essential steps to further refine and generalize the proposal. These efforts could lead to a more inclusive and efficient method for software system development in citizen-centric smart cities.

Artificial intelligence (AI) could play a crucial role in improving this method by facilitating the design of more inclusive and user-friendly participatory techniques. Using AI algorithms, interactive tools could be created and tailored to different citizen profiles, encouraging active participation in the co-creation process. Additionally, AI could be used to interpret behavioral patterns and analyze the outcomes of participatory sessions, providing deeper insights into citizens' needs and preferences. These data could be used to define more precise and effective functionalities for new systems or to improve existing ones, allowing for a more dynamic, user-centered approach to the design of smart urban solutions.

This aligns with the transition from smart cities to more automated or post-smart environments, as described in [67], where artificial intelligence becomes a central actor in urban governance and management. Future research could explore how the method proposed in this study could be adapted to address emerging challenges in post-smart cities, where urban governance and citizen participation are increasingly influenced by AI. Understanding how citizen participation dynamics evolve in these contexts will be crucial for adjusting methods and frameworks for software development, ensuring that post-smart cities remain inclusive and citizen-centered.

**Author Contributions:** Investigation, A.J.R. and J.E.A.; Writing—original draft, A.J.R.; Writing—review & editing, A.J.R., M.A.V. and J.E.A.; Supervision, M.A.V. and J.E.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Science, Technology, and Innovation Fund (FCTeI) of the General Royalties System (SGR) under the project identified by code BPIN 2020000100044, Universidad de Antioquia and Universidad Nacional de Colombia.

**Institutional Review Board Statement:** The study was conducted in accordance with the Declaration of Helsinki and approved by the Comité de Ética de la Universidad Nacional de Colombia Sede Medellín in the ethical approval with code CEMED-06-23 dated 27 February 2023.

**Informed Consent Statement:** Informed consent was obtained from all subjects involved in the study.

**Data Availability Statement:** The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Alexopoulos, C.; Pereira, G.V.; Charalabidis, Y.; Madrid, L. A taxonomy of smart cities initiatives. In Proceedings of the 12th International Conference on Theory and Practice of Electronic Governance, Melbourne, VIC, Australia, 3–5 April 2019; pp. 281–290.
2. Abadía, J.J.P.; Walther, C.; Osman, A.; Smarsly, K. A systematic survey of Internet of Things frameworks for smart city applications. *Sustain. Cities Soc.* **2022**, *83*, 103949. [[CrossRef](#)]
3. Taewoo, N.; Pardo, T. Conceptualizing smart city with dimensions of technology, people, and institutions. In Proceedings of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times, College Park, MD, USA, 12–15 June 2011; pp. 282–291.
4. Paskaleva, K.; Cooper, I. Innovations in Co-Created Smart City Services. In *Setting Foundations for the Creation of Public Value in Smart Cities*; Rodríguez-Bolívar, M.P., Ed.; Springer International Publishing: Cham, Switzerland, 2019; pp. 165–195.
5. Ruhlandt, R.W.S. The governance of smart cities: A systematic literature review. *Cities* **2018**, *81*, 1–23. [[CrossRef](#)]
6. Farias, R.S.; de Souza, R.M.; McGregor, J.D.; de Almeida, E.S. Designing smart city mobile applications: An initial grounded theory. *Empir. Softw. Eng.* **2019**, *24*, 3255–3289. [[CrossRef](#)]
7. Barambones, J.; Moral, C.; Ferre, X.; Villalba-Mora, E. A Scrum-Based Development Process to Support Co-creation with Elders in the eHealth Domain. In *Human-Centered Software Engineering*; Bernhaupt, R., Ardito, C., Sauer, S., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 105–117.

8. Kunttu, I. Developing smart city services by mobile application. In Proceedings of the ISPIM Connects Ottawa, Innovation for Local and Global Impact, Ottawa, ON, Canada, 7–10 April 2019.
9. Jarke, J. Co-creation in Practice I: Co-creating a Digital Neighbourhood Guide (Bremen Osterholz). In *Co-Creating Digital Public Services for an Ageing Society*; Springer: Cham, Switzerland, 2021; Volume 6, pp. 71–116.
10. Jarke, J. Co-Creation in Practice II: Co-creating a Digital Walking Guide (Bremen Hemelingen). In *Co-Creating Digital Public Services for an Ageing Society: Evidence for User-Centric Design*; Springer International Publishing: Cham, Switzerland, 2020; Volume 6, pp. 117–165. [[CrossRef](#)]
11. Santana, E.F.Z.; Chaves, A.P.; Gerosa, M.A.; Kon, F.; Milojevic, D.S. Software Platforms for Smart Cities: Concepts, Requirements, Challenges, and a Unified Reference Architecture. *ACM Comput. Surv.* **2017**, *50*, 1–37. [[CrossRef](#)]
12. Paskaleva, K.; Cooper, I.; Linde, P.; Peterson, B.; Götz, C. Stakeholder Engagement in the Smart City: Making Living Labs Work. In *Transforming City Governments for Successful Smart Cities*; Rodríguez-Bolívar, M.P., Ed.; Springer International Publishing: Cham, Switzerland, 2015; pp. 115–145.
13. Cardullo, P.; Kitchin, R. Being a ‘citizen’ in the smart city: Up and down the scaffold of smart citizen participation in Dublin, Ireland. *GeoJournal* **2019**, *84*, 1–13. [[CrossRef](#)]
14. Rocha, V.; Alves, L.; Vicente, V.; Neto, G.; Kassab, M. A Review on the Adoption of Agile Methods in the Technology Development for Smart Cities. In *Anais Do II Workshop Brasileiro de Cidades Inteligentes*; Sociedade Brasileira de Computação: Porto Alegre, Brazil, 2019; Available online: <https://sol.sbc.org.br/index.php/wbci/article/view/6748> (accessed on 1 December 2024).
15. Simonofski, A.; Sokolvak, D.; Serral, E. Smart City Software: A Review of Development Methodologies and Modelling Languages. In *International Conference on Advanced Information Systems Engineering*; Springer: Cham, Switzerland, 2022; pp. 37–48.
16. Taghi, S.M.J.G.; Ardakani, M. A Reliable Hybrid Software Development Model: CRUP (Crystal Clear & RUP). In *International Conference on Emerging Applications and Technologies for Industry 4.0 (EATI'2020)*; Abawajy, J.H., Choo, K.K.R., Chiroma, H., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 53–69.
17. Pilemalm, S.; Lindell, P.-O.; Hallberg, N.; Eriksson, H. Integrating the Rational Unified Process and participatory design for development of socio-technical systems: A user participative approach. *Des. Stud.* **2007**, *28*, 263–288. [[CrossRef](#)]
18. Lom, M.; Pribyl, O.; Zelinka, T. System engineering for smart cities-hybrid-agile approach in smart cities procurement. In Proceedings of the 20th World Multi-Conference on Systemics, Cybernetics and Informatics, Proceedings (WMSCI), Prague, Czech Republic, 5–8 July 2016; Available online: <https://www.iiis.org/CDs2016/CD2016Summer/papers/SA619KG.pdf> (accessed on 1 December 2024).
19. Castilla, R.; Pacheco, A.; Franco, J. Digital government: Mobile applications and their impact on access to public information. *SoftwareX* **2023**, *22*, 101382. [[CrossRef](#)]
20. Prenner, N.; Unger-Windeler, C.; Schneider, K. Goals and challenges in hybrid software development approaches. *J. Softw. Evol. Process.* **2021**, *33*, e2382. [[CrossRef](#)]
21. Kopec, W.; Nielek, R.; Wierzbicki, A. Guidelines toward Better Participation of Older Adults in Software Development Processes Using a New SPIRAL Method and Participatory Approach. In Proceedings of the 2018 IEEE/ACM 11th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE), Gothenburg, Sweden, 27 May 2018; pp. 49–56.
22. Leong, J.; Yee, K.M.; Baitsegi, O.; Palanisamy, L.; Ramasamy, R.K. Hybrid Project Management between Traditional Software Development Lifecycle and Agile Based Product Development for Future Sustainability. *Sustainability* **2023**, *15*, 1121. [[CrossRef](#)]
23. Cortés, M.; Gil, O. Engagement en ciudades inteligentes. Diseño de un marco de análisis teórico y aplicado para la participación ciudadana. *Gestión Y Análisis De Políticas Públicas* **2018**, *19*, 50–69. [[CrossRef](#)]
24. Pellicano, M.; Calabrese, M.; Loia, F.; Maione, G. Value Co-Creation Practices in Smart City Ecosystem. *J. Serv. Sci. Manag.* **2019**, *12*, 34–57. [[CrossRef](#)]
25. Paskaleva, K.; Cooper, I.; Concilo, G. Co-producing Smart City Services: Does One Size Fit All? In *Smart Technologies for Smart Governments: Transparency, Efficiency and Organizational Issues*; Bolívar, M.P.R., Ed.; Springer International Publishing: Cham, Switzerland, 2018; pp. 123–158.
26. Ruiz, E. Methodologies for a Participatory Design of IoT to Deliver Sustainable Public Services in ‘Smart Cities’. In *Beyond Smart and Connected Governments*; Springer: Berlin/Heidelberg, Germany, 2020; Volume 30, pp. 49–68.
27. Schütz, F.; Heidingsfelder, M.L.; Schraudner, M. Co-shaping the Future in Quadruple Helix Innovation Systems: Uncovering Public Preferences toward Participatory Research and Innovation. *She Ji J. Des. Econ. Innov.* **2019**, *5*, 128–146. [[CrossRef](#)]
28. Costales, E. Identifying sources of innovation: Building a conceptual framework of the Smart City through a social innovation perspective. *Cities* **2021**, *120*, 103459. [[CrossRef](#)]
29. Jacobson, I.; Ng, P.-W.; McMahon, P.E.; Spence, I.; Lidman, S. The essence of software engineering. *Commun. ACM* **2012**, *55*, 42–49. [[CrossRef](#)]
30. Rocha, N.P.; Bastardo, R.; Pavão, J. Citizens’ Participation in the Co-Design of Smart Cities’ Applications, a Scoping Review. In Proceedings of the 10th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion, in DSAI ’22, Lisbon, Portugal, 31 August–2 September 2022; pp. 172–177. [[CrossRef](#)]
31. Kareborn, B.B.; Stahlbrost, A. Living Lab: An open and citizen-centric approach for innovation. *Int. J. Innov. Reg. Dev.* **2009**, *1*, 356. [[CrossRef](#)]

32. Durugbo, C.; Riedel, J.C. Viewpoint–participation–technique: A model of participative requirements elicitation. *Concurr. Eng.* **2013**, *21*, 3–12. [[CrossRef](#)]
33. Fico, G.; Montalva, J.B.; Medrano, A.; Liappas, N.; Mata-Díaz, A.; Cea, G.; Arredondo, M.T. Co-creating with consumers and stakeholders to understand the benefit of Internet of Things in Smart Living Environments for Ageing Well: The approach adopted in the Madrid Deployment Site of the ACTIVAGE Large Scale Pilot. In *EMBECE & NBC 2017*; Eskola, H., Väisänen, O., Viik, J., Hyttinen, J., Eds.; Springer: Singapore, 2018; pp. 1089–1092.
34. Sjøberg, D.I.K.; Dybå, T.; Anda, B.C.D.; Hannay, J.E. Building Theories in Software Engineering. In *Guide to Advanced Empirical Software Engineering*; Shull, F., Singer, J., Sjøberg, D.I.K., Eds.; Springer: London, UK, 2008; pp. 312–336. [[CrossRef](#)]
35. Mendoza-Moreno, M.; González-Serrano, C.; Pino, F.J. Focus Group As A Software Engineering Process: An Experience From The Praxis. *Dyna* **2013**, *80*, 51–60.
36. Arnstein, S.R. A Ladder Of Citizen Participation. *J. Am. Inst. Plan.* **1969**, *35*, 216–224. [[CrossRef](#)]
37. Tadili, J.; Fasly, H. Citizen participation in smart cities: A survey. In Proceedings of the 4th International Conference on Smart City Applications, Casablanca, Morocco, 2–4 October 2019; pp. 1–6.
38. Karvonen, A.; Cugurullo, F.; Caprotti, F. *Inside Smart Cities*; Routledge: London, UK, 2019.
39. Burns, R.; Fast, V.; Levenda, A.; Miller, B. Smart cities: Between worlding and provincialising. *Urban Stud.* **2021**, *58*, 461–470. [[CrossRef](#)]
40. Haklay, M.; Dörler, D.; Heigl, F.; Manzoni, M.; Hecker, S.; Vohland, K. What Is Citizen Science? The Challenges of Definition. In *The Science of Citizen Science*; Springer: Cham, Switzerland, 2021; pp. 13–33. [[CrossRef](#)]
41. Özden, P.; Velibeyoğlu, K. Citizen science projects in the context of participatory approaches: The case of Izmir. *J. Des. Resil. Archit. Plan* **2023**, *4*, 31–46. [[CrossRef](#)]
42. Irwin, A. *Citizen Science: A Study of People, Expertise and Sustainable Development*; Routledge: London, UK, 2002.
43. Hecker, S.; Haklay, M.; Bowser, A.; Makuch, Z.; Vogel, J.; Bonn, A. Innovation in open science, society and policy–setting the agenda for citizen science. In *Citizen Science: Innovation in Open Science, Society and Policy*; UCL Press: London, UK, 2018; pp. 1–23.
44. Weber, K.; Pallas, F.; Ulbricht, M.-R. Challenges of citizen science: Commons, incentives, organizations, and regulations. *Am. J. Bioethics* **2019**, *19*, 52–54. [[CrossRef](#)] [[PubMed](#)]
45. Guenduez, A.A.; Mettler, T.; Schedler, K. Citizen participation in smart government: A conceptual model and two IoT case studies. In *Beyond Smart and Connected Governments*; Springer: Cham, Switzerland, 2020; pp. 189–209.
46. Simonofski, A.; Asensio, E.S.; De Smedt, J.; Snoeck, M. Citizen Participation in Smart Cities: Evaluation Framework Proposal. In Proceedings of the 2017 IEEE 19th Conference on Business Informatics (CBI), Thessaloniki, Greece, 24–27 July 2017; Volume 1, pp. 227–236.
47. Bastos, D.; Fernández-Caballero, A.; Pereira, A.; Rocha, N.P. Smart City Applications to Promote Citizen Participation in City Management and Governance: A Systematic Review. *Informatics* **2022**, *9*, 89. [[CrossRef](#)]
48. Hognogi, G.-G.; Meltzer, M.; Alexandrescu, F.; Ștefănescu, L. The role of citizen science mobile apps in facilitating a contemporary digital agora. *Humanit. Soc. Sci. Commun.* **2023**, *10*, 1–16. [[CrossRef](#)]
49. Arlati, A.; Rödl, A.; Kanjaria-Christian, S.; Knieling, J. Stakeholder participation in the planning and design of nature-based solutions. Insights from CLEVER cities project in Hamburg. *Sustainability* **2021**, *13*, 2572. [[CrossRef](#)]
50. Castelnovo, W. Coproduction and Co-creation in Smart City Initiatives: An Exploratory Study. In *E-Participation in Smart Cities: Technologies and Models of Governance for Citizen Engagement*; Springer International Publishing: Cham, Switzerland, 2018; Volume 34, pp. 1–20. [[CrossRef](#)]
51. Alford, J. The Multiple Facets of Co-Production: Building on the work of Elinor Ostrom. *Public Manag. Rev.* **2014**, *16*, 299–316. [[CrossRef](#)]
52. Prahalad, C.K.; Ramaswamy, V. Co-creation experiences: The next practice in value creation. *J. Interact. Mark.* **2004**, *18*, 5–14. [[CrossRef](#)]
53. Robert, G.; Donetto, S.; Williams, O. Co-designing healthcare services with patients. In *The Palgrave Handbook of Co-Production of Public Services and Outcomes*; Springer: Cham, Switzerland, 2021; pp. 313–333.
54. Farr, M. Co-Production and Value Co-Creation in Outcome-Based Contracting in Public Services. *Public Manag. Rev.* **2016**, *18*, 654–672. [[CrossRef](#)]
55. Torvinen, H.; Haukipuro, L. New roles for end-users in innovative public procurement: Case study on user engaging property procurement. *Public Manag. Rev.* **2018**, *20*, 1444–1464. [[CrossRef](#)]
56. Margarita, A.; Isabel, F.; Eleni, K.; Meia, W. Co-creation Techniques and Tools for Sustainable and Inclusive Planning at Neighbourhood Level. Experience from Four European Research and Innovation Projects. In *Advances in Mobility-as-a-Service Systems*; Nathanail, E.G., Adamos, G., Karakikes, I., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 562–572.
57. Kairy, D.; Mostafavi, M.A.; Blanchette-Dallaire, C.; Belanger, E.; Corbeil, A.; Kandiah, M.; Wu, T.Q.; Mazer, B. A Mobile App to Optimize Social Participation for Individuals with Physical Disabilities: Content Validation and Usability Testing. *Int. J. Environ. Res. Public Health* **2021**, *18*, 1753. [[CrossRef](#)]
58. Yu, J.; Wen, Y.; Jin, J.; Zhang, Y. Towards a service-dominant platform for public value co-creation in a smart city: Evidence from two metropolitan cities in China. *Technol. Forecast. Soc. Chang.* **2018**, *142*, 168–182. [[CrossRef](#)]
59. Durugbo, C.; Pawar, K. A unified model of the co-creation process. *Expert Syst. Appl.* **2014**, *41*, 4373–4387. [[CrossRef](#)]
60. Osterwalder, A.; Pigneur, Y.; Smith, A.; Greg, B.; Papadacos, P. *Value Proposition Design*, 1st ed.; Wiley: Hoboken, NJ, USA, 2015.

61. Calabrese, J.; Esponda, S.; Boracchia, M.; Pesado, P. Scrum Towards IRAM-ISO 9001:2015. Integrating Documentation Required. In *Computer Science—CACIC 2018*; Patricia, C.P., Aciti, Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 183–196.
62. Tufail, H.; Qasim, I.; Masood, M.F.; Tanvir, S.; Butt, W.H. Towards the selection of Optimum Requirements Prioritization Technique: A Comparative Analysis. In *Proceedings of the 2019 5th International Conference on Information Management (ICIM)*, Cambridge, UK, 24–27 March 2019; pp. 227–231.
63. Tritter, J.Q.; Landstad, B.J. *Focus Groups. Qualitative Research in Health Care*; BMJ Books: London, UK, 2006; pp. 57–66.
64. Muijeen, K.; Kongvattananon, P.; Somprasert, C. The key success factors in focus group discussions with the elderly for novice researchers: A review. *J. Health Res.* **2020**, *34*, 359–371. [[CrossRef](#)]
65. Salazar, A.A.B. Modelo para la Definición Unificada de la Práctica como Constructo Teórico en Ingeniería de Software. 2019. Available online: <https://dialnet.unirioja.es/servlet/tesis?codigo=330276> (accessed on 1 December 2024).
66. Ricardo, A.J.; Vallejo, M.A.; Aedo, J.E. Co-Creating Products and Services for Smart Cities. In *Proceedings of the 2023 IEEE Colombian Conference on Communications and Computing (COLCOM)*, Bogota, Colombia, 26–28 July 2023; pp. 1–5.
67. Cugurullo, F.; Caprotti, F.; Cook, M.; Karvonen, A.; Mcguirk, P.; Marvin, S. The rise of AI urbanism in post-smart cities: A critical commentary on urban artificial intelligence. *Urban Stud.* **2024**, *61*, 1168–1182. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.