World Journal of Information Systems 3(4) 27-53
DOI: https://doi.org/10.17013/wjis.v3i4.70

Article

From Governance Principles to Architectural Primitives: Designing and Evaluating a Microservices-Based Smart Governance Platform for a Wellness Tourism Ecosystem

Efraín Ortiz-Pabón[1,*], Nestor Armando Nova-Arévalo[2], Lady Fernández-Mora[3], Daniela Flórez-Villamil[4], Sara Juliana Peña-Gómez[4], Juan David Hernández-Rodríguez[4], María Andrea Méndez-Beltrán[4]

[1]Pontificia Universidad Javeriana, Departamento de Ingeniería de Sistemas, Bogotá, Colombia
[2]Pontificia Universidad Javeriana, Departamento de Ciencia de la Información, Bogotá, Colombia
[3]Instituto Tecnológico de Costa Rica (TEC), Campus San Carlos, Alajuela, Costa Rica
[4]Pontificia Universidad Javeriana, Carrera de Ingeniería de Sistemas, Bogotá, Colombia

***** Correspondence: efrain.ortiz@javeriana.edu.co

(Article history: Received September 03, 2026; Received in revised form October 30, 2026; Accepted November 28, 2026; Available online December 15, 2026)

Abstract: The growing importance of Information and Communication Technologies (ICTs) in tourism underscores the need for governance models that integrate strategic processes with technological infrastructures. Although digital platforms are widely credited with operationalizing governance principles such as transparency, accountability and participation, the mechanisms through which architectural decisions enact those principles remain largely unspecified. This article presents the design and evaluation of a smart governance platform for a wellness tourism ecosystem [1], developed through a hybrid methodology combining Design Science Research with the Waterfall model and Scrum. The study makes three contributions. First, it proposes an explicit mapping between six governance principles and the architectural primitives that enact them — bounded contexts with private data stores, asynchronous event propagation, role-based access control, and a consolidated analytical store — together with the observable evidence each primitive produces. Second, it documents the resulting microservices-based artifact as a reference decomposition, stating the alternatives rejected and the trade-offs accepted. Third, it reports an evaluation combining architectural analysis with functional and load testing, identifying sensitivity points and residual risks alongside confirmed strengths in scalability and reliability. The findings show that governance principles are not satisfied by digital platforms

[1] This article is an extended version of the paper "Designing a Smart Corporate Governance Prototype for Wellness Tourism Ecosystems: An ICT-Driven Approach", presented at the 9th International Conference on Information Technology & Systems (ICITS'26).

© 2026 ITMA. All rights reserved.

in general, but by specific and traceable architectural choices — notably the reconciliation of transparency and autonomy through the propagation mechanism rather than through access-control policy.

Keywords: Smart Tourism, Smart Governance, Digital Innovation, Shared Value, Microservices Architecture, Software Architecture, Design Science Research

1. Introduction

The global slowdown experienced by the tourism sector during the pandemic exposed the sector's technological vulnerabilities, particularly in Latin America. This crisis shed light on companies' lack of preparedness to offer tourism value propositions leveraging information and communications technologies (ICTs) and emerging digital technologies, focusing on smart tourism. This type of tourism refers to the integrated effort of collecting and utilizing large volumes of data from diverse sources to create proposals that enhance the tourist experience, improve efficiency, and promote sustainability **[1]**. These developments rest on an infrastructure of data-generating sensors, large-scale storage, new forms of connectivity and Internet of Things transmission, combined with analytics, immersive interfaces, cloud computing and artificial intelligence — an infrastructure described in detail in Section 2.1.

In line with the above, the collaboration, interdependence, coordination, interconnection, synchronization, and alignment required among the numerous actors involved in a tourism ecosystem revealed another major weakness in the sector: the lack of smart governance models useful for bridging the gap between the physical and digital worlds. This is what [2] refers to as the development of digital innovations or innovative technological solutions that integrate digital technologies, allowing companies to share data and join shared value platforms useful for updating and enhancing their business models. Given that the relationships between ecosystem actors are dynamic and complex, a smart tourism ecosystem is the most comprehensive way to describe the conceptual foundations of this new tourism model—smart tourism—which is based on the principle that nothing works in isolation, but rather evolves through interaction within the ecosystem [3]. Smart tourism is characterized by multiple interconnected components and layers, supported by ICTs, which give rise to the development of smart destinations, experiences, and businesses [1].

This article presents the design, architectural specification and evaluation of a platform ecosystem [4], [5] that, using a microservices-based architecture, operationalizes smart governance principles and supports shared value creation among the actors in a wellness tourism ecosystem. The prototype acts as a digital infrastructure to coordinate autonomous actors, aligning with the theory of digital ecosystems. This initiative, a consortium collaboration between universities in Colombia and Costa Rica, lays the groundwork for a broader system that could function as a smart governance mechanism for the tourist

© 2026 ITMA. All rights reserved.

destination of La Fortuna, Costa Rica. The artifact was developed through design science research as an integrative methodology, combined with the waterfall methodology for solution development and Scrum during the implementation phase.

The literature on smart tourism governance converges on a set of normative principles—transparency, accountability, participation, collaboration, autonomy, and evolvability—and on the claim that digital platforms are the appropriate instrument for enacting them [3], [6], **[7]**. What this literature seldom specifies is the mechanism: which concrete architectural decision produces which governance property, and what observable evidence demonstrates that the property has actually been enacted rather than merely asserted. The gap is not only conceptual. Destination management organizations invest in digital platforms on the assumption that governance benefits follow automatically from digitalization, and small providers are asked to contribute data on the same assumption. Neither assumption is warranted. A platform that consolidates operational records under a shared schema delivers visibility to whoever administers the schema while dissolving the informational autonomy of the actors who supplied the data—an outcome that reproduces, rather than corrects, the asymmetries that motivated the intervention. Whether a platform enacts governance principles or merely claims them is therefore decided at the level of architecture, and it is decided early.

Accordingly, this study addresses three research questions:

- RQ1. Which architectural primitives of a microservices-based platform enact the governance principles attributed to smart tourism ecosystems, and through which observable evidence?
- RQ2. What design trade-offs arise when these primitives are instantiated under the resource constraints typical of a Latin American destination, and how do those trade-offs affect the governance properties obtained?
- RQ3. To what extent does the resulting architecture satisfy the quality attributes required for multi-actor coordination, as assessed through a structured architectural evaluation?

In answering these questions, the article makes three contributions. First, it proposes an explicit mapping between six smart governance principles and the architectural primitives that enact them, together with the observable evidence by which enactment can be verified; this renders the operationalization claim testable rather than rhetorical. Second, it documents the resulting artifact as a reference decomposition, making explicit not only the design adopted but the alternatives rejected, the criteria that discriminated between them, and the costs accepted—information ordinarily absent from architectural descriptions of governance platforms and without which such descriptions cannot be assessed or reused. Third, it reports a structured evaluation that identifies not only what the architecture achieves but where it remains sensitive and at risk, on the premise that a design-science artifact should be evaluated against its stated objectives rather than presented as complete.

© 2026 ITMA. All rights reserved.

The article is structured as follows: Section 2 presents the conceptual foundations, including definitions of smart tourism, smart governance, digital innovation, shared value, and microservices architecture, and closes by deriving the governance–architecture mapping that guides the design. Section 3 details the methodology, combining Design Science Research (DSR), Waterfall, and Scrum. Section 4 specifies the architecture of the artifact through a C4 decomposition, makes explicit the design decisions and trade-offs adopted, and reports the evaluation. Section 5 discusses the findings and outlines the main conclusions, limitations, and future agenda.

2. Conceptual foundation

The underlying body of knowledge on which this paper is based involves the linking of five key concepts. The first four—smart tourism, smart governance, digital innovation, and shared value—establish what a governance arrangement in a tourism ecosystem is expected to achieve. The fifth, microservices architecture, establishes how the systems that support such arrangements are structured. Section 2.6 then connects the two, which is the conceptual move on which the design reported in this article rests.

2.1. Smart Tourism

Smart tourism is defined as the integrated effort to collect and use large amounts of data, from diverse sources, to create value propositions that improve the tourist experience, efficiency and sustainability [1]. This concept, which has been frequently used in the literature, is applied to characterize the growing dependence of tourism destinations, their industries, and their tourists on emerging forms of Information and Communication Technologies (ICTs) because they allow the transformation of massive amounts of data into value propositions [1]. Smart tourism is based on the paradigm of high-tech service production, characterized by adopting the principles of interoperability, virtualization, decentralization, service orientation, data analytics, and modularity, which are key aspects of Industry 4.0 [8]. This approach has driven full automation in the production and delivery of tourism goods and services, leading to potential technological disruptions in the sector [9]. The term "smart" refers to the application of a complex amalgamation of technological, economic, and social developments driven by new digital technologies based on the use of data-generating sensors, the storage of large volumes of data (Big Data), access to complementary open data (e.g., originating from the web, social networks), new forms of connectivity, the capture and transmission of data using the Internet of Things - IoT, technologies such as RFID (an acronym for radio-frequency identification), NFC (an acronym for Near Field Communication), associated data analytics to extract value, virtual, augmented, and hybrid reality to design new experiences, cloud computing to store and process large volumes of data, and, more recently, artificial intelligence - AI to add value to them; which in an interconnected, synchronized, and concerted manner allow us to close the

© 2026 ITMA. All rights reserved.

gap between the physical world and the digital world. Unlike its predecessor, e-tourism, which focused primarily on the digital world, smart tourism encompasses both the digital and physical worlds. Smart tourism is characterized by the fact that it encompasses multiple components and layers supported by ICTs that give rise to the development of smart destinations, smart experiences, and smart businesses [1].

Two of the Industry 4.0 principles enumerated above—decentralization and modularity—are not merely descriptive of smart tourism but prescriptive for the systems that support it. They reappear in Section 2.5 as constitutive properties of microservice architectures, and this correspondence is not incidental: the reason smart tourism is characterized by multiple components and layers is that the ecosystem it describes is itself composed of autonomous actors that no single organization owns or controls. A system intended to support such an ecosystem inherits that condition as a design constraint rather than adopting it as a stylistic preference.

2.2 Smart Governance

Smart governance is defined as the ability to apply digital technologies and intelligent activities in information processing and decision-making [10]. When applied to tourism, its focus shifts to smart tourism, which sets the requirements for destinations to integrate physical, social, and technological infrastructure in order to enhance the experiences of both tourists and residents, while adapting to the evolving needs of destination stakeholders [9], [3]. The concept of smart governance does not depart from the traditional foundations of corporate governance theory; rather, it builds upon them by incorporating the intelligent use of ICT to improve decision-making processes [11], [3]. Although technological resources are considered a necessary condition within the structure of the smart approach, social capital combined with governance remains a crucial factor for the success of a smart destination. From this perspective, smart governance can be defined through a sociotechnical lens, aligning technological potential with new forms of collaboration between local governments and citizens, with the goal of addressing challenges based on sustainability principles [12],[6],[3]. Smart governance is essential for promoting access to information, transparency, and public participation in decision-making processes involving various stakeholders. This ensures that governance becomes transparent, accountable, collaborative, and participatory—facilitating knowledge sharing and transfer [13], [14], [7]. In this context, the tourist is no longer a passive subject but rather becomes an active participant, as smart tourism views the tourist as a co-creator of experiences. To fully understand the scope of smart governance, it is necessary to adopt an ecosystem perspective. This approach encompasses elements such as tourists, governments, natural environments, tourism service providers, information exchange centers, residents, suppliers from other industries, media outlets, destination management organizations, and digital technologies, among others. In other words, an effective smart governance model today must consider most—if not all—of the actors involved in the tourism ecosystem. However, [3] proposes organizing the smart tourism

© 2026 ITMA. All rights reserved.

ecosystem around the interaction of four core elements: digital technologies, consumers (tourists and residents), businesses (tourism-related and other sectors), and the tourist destination (governed by destination management organizations and public authorities). Additional actors such as universities, communities, and civil society also play key roles, all of them interconnected through intensive use of technology.

The principles enumerated in this literature—transparency, accountability, collaboration, participation—are stated as properties that governance arrangements ought to exhibit, and the sociotechnical framing indicates that technology is expected to contribute to their realization. What remains underspecified is the form of that contribution. The statement that a platform promotes transparency admits at least two readings: that the platform makes information available to actors who would otherwise lack it, or that the platform makes the conduct of other actors legible. These are different arrangements with different distributional consequences, and they are produced by different technical designs. The same ambiguity affects participation, which may denote the capacity to contribute data or the capacity to influence decisions, and accountability, which may denote traceability of actions or answerability for them. Section 2.6 addresses this by requiring that each principle be tied to a mechanism and to evidence, so that the reading being claimed is determinate.

2.3 Digital Innovation

Digital innovation (DI) is defined as the combination of physical and digital products to create new products, services or processes, through the design, development, implementation and exploitation of digital technological artifacts in an organization [15]. From the perspective of [17] DI is conceived as a set of routine and interdependent social interactive activities carried out by humans, mediated by digital technology, and goal-oriented. They can arise from the opportunities that reside in digital ecosystems [16], in the digital transformation processes undertaken by organizations, in the opportunities derived from the application of ICT and new digital technologies. Another conceptual perspective indicates that digital innovation refers to the organizational adoption and diffusion of new IT-enabled processes, products, and services [17], [18]. In this conceptualization, innovation refers to the adoption of an existing IT artifact that is new to an organization and is presumably driven by various technological, organizational, and environmental characteristics. Related concepts to IT innovation include IT diffusion and assimilation [15]. From the point of view of its application to sectors such as tourism, it usually refers to the creation of value through the strategic and transformative use of ICT and digital technologies by making it smarter, because they consider both technological and organizational, contextual and sustainable aspects. In this logic, DI can be seen as a dynamic and multidimensional process that transforms not only products or services, but also the ways in which organizations operate, cooperate, coordinate while competing.

As DI allows proposing IT solutions that integrate digital technologies to support the digitalization of businesses in different sectors such as banking, healthcare, manufacturing,

© 2026 ITMA. All rights reserved.

retail, tourism, among others [2], they can be summarized as the process of combining resources, especially digital knowledge and skills to promote the development of organizational capabilities and generate novel solutions [19]. It is essential to give body to new strategic management constructs such as smart governance and in this way help improve decision-making, coordination and the evolution of tourism ecosystems. This process involves interactive social activities mediated by digital technology and driven by specific goals [16], where collaboration between government, the private sector, and civil society converges and becomes necessary.

Understanding digital innovation as a process of interdependent social activities mediated by technology, rather than as the introduction of an artifact, carries a consequence for how the present work should be read. The platform described in Sections 4 and 5 is not itself the innovation; the innovation is the reconfiguration of coordination among ecosystem actors that the platform makes possible. This distinction matters for evaluation. An artifact assessed only on its technical properties may satisfy every quality attribute while leaving the coordination problem untouched—if, for instance, the information it exposes is not the information actors require in order to act differently. It also matters for design, since it implies that the criteria by which architectural alternatives are chosen must include their effect on the interaction among actors, and not only their effect on system properties such as performance or maintainability. Section 4.7 applies this criterion explicitly.

2.4 Shared Value

The concept of shared value refers to operational policies and practices that enhance a company's competitiveness while improving the economic and social conditions of the communities in which it operates. It involves creating economic value in a way that also generates value for society by addressing its needs and challenges. Fundamentally, it represents a reconnection between business success and social progress [20]. This distributed value creation perspective finds its natural embodiment in platform ecosystems, where modular architectures and governance mechanisms enable the coordination of autonomous actors to generate specific complementarities and shared value [5], [4]. They are also understood as economic communities or integrated and holistic technological systems, whose structure is made up of organizations, individuals from the business world, and subsystems that lead to the consolidation of networks characterized by the generation of interaction relationships [21]. Value creation is a long-recognized idea in business, where profit is the revenue earned from customers less the costs incurred. Companies can create economic value by creating social value.

There are three distinct ways to do this: rethinking products and markets, redefining productivity in the value chain, and building supporting industrial clusters at company locations. Each of these is part of the virtuous cycle of shared value; improving value in one area leads to opportunities in others [20]. Precisely, digital platforms are a phenomenon that challenges incumbents by changing the way we consume and provide digital products and

© 2026 ITMA. All rights reserved.

services. While traditional businesses create value within the boundaries of a company or supply chain, digital platforms utilize an ecosystem of autonomous agents to co-create value [4]. The tourism sector is no exception; on the contrary, offerings and consumption patterns have transformed as businesses adopt digital transformation processes. With the advent of smart tourism, digital innovations became apparent, and smart tourist destinations and regions began to develop. Tourism governance models also evolved toward smart tourism destination governance models supported by ICTs and new digital technologies. This has led to the consolidation of dynamic tourism ecosystems where collaboration between multiple stakeholders implies a fundamental change in the way tourism value and experiences are created, exchanged, consumed, and shared. The above perspective implies an opportunity for digital innovators because transforming data into meaningful information that supports value creation by the different actors that make up the ecosystem becomes fundamental, therefore, ecosystems can be perceived as a structure, which helps to outline a particular context where multiple actors, dynamically organized around a shared value purpose, co-create value mutually [22] and digital platforms become the indicated mechanism when based on software, extensible code bases that provide basic functionalities, complemented with modular services[4].

Shared value in a platform ecosystem is nonetheless conditional on the terms under which actors participate, and those terms are set in part by the platform itself. An actor that contributes operational data to a shared infrastructure incurs a cost—exposure of information about its own activity—against an expected benefit in the form of access to aggregate information it could not produce alone. Where the exposure is asymmetric, so that some actors observe others without being observed in turn, or where the aggregate benefit accrues disproportionately to whoever administers the infrastructure, participation ceases to be rational for the smaller contributors and the ecosystem does not form. This is a recurrent failure mode in destinations composed predominantly of micro and small enterprises, as documented for the Costa Rican cluster that motivates the present case [19]. The design implication is that the conditions under which shared value is attainable are partly architectural: they depend on whether the platform can deliver aggregate visibility without requiring actors to surrender control over their operational records. Section 2.6 treats this as a design requirement rather than as a policy matter.

2.5 Microservices Architecture

A microservice is an independently deployable component of bounded scope that supports interoperability through message-based communication. Microservice architecture is a style of engineering highly automated, evolvable software systems made up of capability-aligned microservices [23]. A microservices-based system encompasses all aspects of the organization related to the application it produces. This means that the organization's structure, the people who work within it, their work methodology, and the results they generate are all important factors within the system [23]. Microservices design consists of

© 2026 ITMA. All rights reserved.

five parts: Service, Solution, Process and Tools, Organization, and Culture [23]. Services are the basic building blocks upon which the entire system is built. When designing a solution architecture, decisions are constrained by the need to coordinate all the inputs and outputs of multiple services. A microservices system includes tools and processes related to software development, code deployment, maintenance, and product management. Good service design stems from good organizational design; from a microservices perspective, organizational design includes structure, authority, granularity, and team composition. Culture can be broadly defined as a set of values, beliefs, or ideals shared by all employees within an organization.

Three properties of this architectural style are consequential for the argument developed here. The first is bounded scope with private state: each service owns its data and no other service may read it directly, so that integration occurs exclusively through published interfaces. The second is message-based interoperability, which permits communication to be asynchronous, so that the availability of a consumer is not a precondition for the operation of a producer. The third is capability alignment, which locates service boundaries at divisions of responsibility rather than at technical layers, with the consequence that the units of independent change coincide with the units of organizational concern.

The framing offered by [23]—that a microservices system encompasses the organization producing it, including its structure, distribution of authority, and granularity—extends naturally beyond the development organization. In the setting considered here, the relevant organization is not a single firm but an ecosystem of autonomous actors, and the correspondence between service boundaries and organizational boundaries carries the same force: a decomposition that cuts across the divisions of authority in the ecosystem will require coordination that the ecosystem cannot supply, while a decomposition aligned with them permits each actor's capability to evolve at its own pace. The five-part design frame proposed by [23] therefore admits a reading in which Organization denotes the ecosystem rather than the development team, and this reading is the one adopted in Section 4.

2.6 From governance principles to architectural primitives

The five constructs reviewed above are usually treated as parallel literatures. Smart governance research specifies what a governance arrangement ought to achieve; microservices research specifies how systems ought to be decomposed. The connection between the two is generally left implicit, expressed as the assertion that modular digital platforms support governance principles [3],[4],[5]. This assertion is not wrong, but it is unfalsifiable as stated: any platform can be said to support transparency, and no criterion is offered for distinguishing platforms that enact a principle from platforms that merely claim it.

We therefore adopt a stricter formulation. A governance principle is enacted by an architecture when three conditions hold jointly: (i) a specific architectural primitive is present in the design; (ii) that primitive causally produces a state of affairs corresponding to the

© 2026 ITMA. All rights reserved.

principle; and (iii) the state of affairs leaves observable evidence that an external party could inspect. Under this formulation, transparency is not achieved by having a dashboard, but by the fact that every state-changing operation in every bounded context emits an event to a shared broker, that these events are consolidated into an analytical store independent of the operational stores, and that the resulting aggregates are queryable by actors who do not own the originating service.

The distinction has a direct architectural consequence, which can be seen most clearly in the relation between transparency and autonomy. Under a monolithic design with a shared schema, transparency would be obtained by granting cross-domain read access—which simultaneously destroys autonomy, since each actor could then read the operational state of the others. Under an event-propagation design, transparency is obtained on the aggregated projection while autonomy is preserved on the operational data, because no service can read another service's store. Two principles ordinarily in tension are reconciled by the choice of propagation mechanism rather than by policy. The significance of this is that the reconciliation holds structurally: it does not depend on an access-control rule being correctly configured, nor on an administrator refraining from exercising a capability the system affords. Governance properties secured by structure are more robust than those secured by configuration, and the difference is precisely what the third condition above is intended to detect, since a structurally enacted property leaves evidence in the design itself and not merely in a policy document.

This formulation also clarifies what the framework does not claim. Establishing that an architecture enacts a governance principle does not establish that the governance arrangement is effective, equitable, or adopted. The mapping is a necessary-condition analysis: it identifies what must be true of the design for the principle to be attainable at all, leaving open whether actors will in fact use the resulting capability, and whether doing so will produce the coordination outcomes sought. Section 5 returns to this boundary when discussing the limits of what the present evaluation can support.

Table 1 formalizes the reasoning for the six governance principles that recur in the smart tourism governance literature [3],[6],[13],[14],[7], stating for each the ecosystem requirement it imposes, the architectural primitive that enacts it, its instantiation in the present artifact, and the evidence by which enactment can be verified. The six are not independent: as shown above, transparency and autonomy are jointly determined by the propagation mechanism, and accountability presupposes the identity mechanism on which participation and collaboration also rely. The table should therefore be read as a set of interacting requirements on a single design rather than as a checklist of separable features. It constitutes the design rationale for the architecture presented in Section 4 and the evaluation criterion applied in Section 4.7.

Table 1. Mapping between smart governance principles and architectural primitives

© 2026 ITMA. All rights reserved.

Governance principle	Ecosystem requirement		Architectural primitive	Instantiation in the artifact	Observable evidence
Transparency	Actors must gain visibility over demand, supply and territorial behaviour they do not themselves own		Event propagation to a consolidated analytical store, decoupled from operational stores	AMQP events published by each microservice; Python consumer; wellness-long-persistence-db (MySQL)	Provider and administrator dashboards report aggregates whose source services the querying actor does not control
Accountability	Every state change must be attributable to an identified role		Role-based access control combined with an append-only event record	User Management API issuing role-scoped credentials; timestamped events emitted per operation	Each recorded event carries actor identity, role and timestamp, evaluated at the point of commitment
Participation / co-creation	Tourists must act as producers of ecosystem data, not only as consumers of services		Elevation of user preferences to a first-class bounded context rather than an attribute of the user record	Preferences API with its own store and its own event stream	Preference events measurably alter recommendations and provider-facing demand indicators
Collaboration	Actors must share information without ceding ownership of their operational data		Asynchronous message-based integration between bounded contexts; absence of a	RabbitMQ (AMQP) as the sole inter-context channel	No cross-service database references exist; each service retains an independent SQLite store

© 2026 ITMA. All rights reserved.

			shared schema		
Autonomy / decentralization	Each actor's capability must evolve and fail without cascading to others		Independent deployability of capability-aligned services	Provider Services API, User Management API and Preferences API deployed independently (Spring Boot)	Unavailability of one service degrades its capability only; the broker buffers pending events
Evolvability / openness	New actors and capabilities must join over time without renegotiating existing integrations		Additive subscription to published topics under stable contracts	New consumers subscribe to existing exchanges without modification of producing services	Addition of the analytics consumer required no change to the three producing microservices

3. Methodology

To design the prototype of a smart governance platform for wellness tourism, we adopted a hybrid approach combining Design Science Research (DSR), the Waterfall model, and Scrum. DSR structured the research process, guiding problem identification, objective definition, and the theoretical grounding of the artifact. The Waterfall model ensured rigor in requirements analysis, system design, and documentation, while Scrum introduced flexibility during implementation through iterative cycles and continuous feedback from domain experts. This combination was necessary because the project required both scientific rigor and adaptability to the dynamic needs of tourism ecosystems. The hybrid methodology not only supported the development of the functional prototype but also generated practical insights into how methodological integration can enhance ICT-driven governance models in the tourism sector. Following the iterative DSR research process, the following steps are taken: problem identification, definition of solution objectives, design and development, demonstration, and evaluation and communication of results.

3.1 Problem Identification

The challenge consisted in developing the prototype of a shared-value platform aimed at facilitating better coordination among the actors of the tourism ecosystem through the exchange of data oriented toward the generation of smart governance. Similar approaches

© 2026 ITMA. All rights reserved.

have been discussed in the context of challenge-based learning [24], which emphasizes the role of problem-driven design in higher education. However, in this study the focus is placed on applying these principles to the development of a governance prototype for the tourism sector Coordination, from the perspective of smart governance of tourist destinations, involves the use of Information and Communication Technologies (ICTs) and the integration of emerging digital technologies that act as enablers for capturing, storing, processing, and communicating valuable information among the different actors that make up the ecosystem: tourists, tourist attractions, travel agencies, transportation systems, tour guides, chambers of commerce, government, academia, and other stakeholders acting as regulators, trainers, and value creators. Smart governance should enable stakeholders to develop a more holistic understanding of the importance of sustainability and competitiveness for the sector, the region, tourist destinations, and the community.

3.2 Solution Objectives

The objective is to design the functional prototype of the platform to facilitate smart governance and generate shared value among the stakeholders of the tourism ecosystem. The solution should enable coordination among stakeholders. The project must cover the definition of functional and non-functional requirements, propose an architecture, and validate the resulting artifact.

The scope of validation pursued in this study is architectural and technical. It comprises three components: an assessment of the architecture against prioritized quality-attribute scenarios using the Architecture Tradeoff Analysis Method (ATAM); functional verification of the implemented services through unit and integration testing; and behavioural testing under simulated concurrent load to establish capacity and degradation characteristics. Complementing these, the architectural primitives identified in Table 1 are examined for the observable evidence each is expected to produce, so that the enactment of the governance principles can be assessed against the criterion established in Section 2.6 rather than assumed.

Participatory evaluation with end users — qualitative validation of usability, adoption and trust with tourists, providers and destination management representatives — falls outside the scope defined here and is set out as the first item of the research agenda in Section 5. This delimitation is deliberate: the architectural properties on which the governance claim rests can be established independently of adoption, whereas evidence of adoption presupposes an architecture whose properties have already been demonstrated.

3.3 Design and development

This part of the process was carried out using the Waterfall methodology, which enabled detailed planning and comprehensive documentation prior to the implementation of the prototype. Agile methodologies fostered flexibility and adaptability through rapid iterations and continuous testing with wellness tourism experts. Before developing the prototype, an

© 2026 ITMA. All rights reserved.

analysis of the environment of the selected tourist destination was conducted, and secondary sources such as [25] were consulted to gain a better understanding of the problem context. Interviews were conducted with key stakeholders of the ecosystem, and the results were used to refine the definition of functional and non-functional requirements. A Product Backlog was created to facilitate the management and prioritization of requirements, allowing for greater flexibility in planning and adapting the system to potential changes in the challenge's needs.

3.4 Data sources and research protocol

Design science research derives its validity from a relevance cycle that grounds the artifact in a real problem environment and a rigor cycle that grounds it in an existing knowledge base [26].

Sections 3.1 to 3.3 described the activities performed; this subsection reports the evidentiary basis on which they rested, so that the requirements presented in Section 4.1 and the design decisions consolidated in Section 4.7 can be traced to their sources.

Three classes of input informed the design. Secondary evidence about the destination established the conditions under which the architecture had to be viable: a study of digital maturity across six cantons of Alajuela, Costa Rica, characterized the technological capabilities and the predominance of micro and small enterprises in the cluster [20], and prior work on information system design for smart tourism destination transformation supplied the domain framing [25].

This is the reason the design privileges primitives that a small technical team can instantiate over primitives that presuppose institutional data infrastructure. Primary evidence was elicited through semi-structured interviews with key stakeholders of the ecosystem, whose accounts of the information they require but do not obtain were translated into candidate requirements and consolidated into the Product Backlog. Finally, wellness tourism experts reviewed each increment at the close of the Scrum cycles, providing formative feedback that operated on the working artifact rather than on the problem statement — a distinction the hybrid methodology makes visible and a purely sequential process would obscure.

The knowledge base contributed on two fronts: the smart tourism governance literature [3], [6],[13],[14],[7], from which the six principles constituting the design objectives were drawn, and the microservices literature [23], from which the architectural primitives and the decomposition criterion were taken. The correspondence between the two is established in Table 1 and applied throughout Section 4.

Two boundaries of this protocol should be stated at the point of use. The interviews served requirements elicitation and were not designed to support inference about the ecosystem beyond the participating actors; no claim of representativeness is made. And the evidence they yielded informs the specification of the artifact, not its validation: the scope of validation is defined in Section 3.2 and reported in Section 4.7.

© 2026 ITMA. All rights reserved.

4. Artifact design and evaluation

This section specifies the artifact. Rather than presenting the design as an undifferentiated set of diagrams, we document it following the C4 model, which organizes architectural description into progressively detailed levels — system context, containers and components. Section 4.2 establishes the overall runtime organization; Sections 4.3 to 4.5 refine it through the three levels. The code level is omitted, as it carries no governance-relevant information. Each level is presented together with the design decision it embodies, so that the architecture can be read against the mapping established in Table 1.

4.1 Requirements

The project included user interfaces, the design of the microservices architectural pattern, and addressed software quality attributes according to ISO 25010. It also covered the justification of the selected technologies and the corresponding design diagrams. The list of functional requirements agreed upon with the client can be found in Table 2. The non-functional requirements can be found in Table 3.

Table 2. Functional Requirements Summary

Module	Description
Authentication and Registration	**Tourist Registration:** The system must allow the registration of new tourists, including personal information and access credentials, as long as their email address has not been previously registered.
	Supplier Registration: The system must allow the registration of new suppliers, including personal information and access credentials, as long as their email address has not been previously registered.
	Login: The system must allow registered users to authenticate using a username and password.
	Password Recovery: The system must allow registered users to recover their password using their email address.
Visualization and Metrics	**View Supplier Dashboard:** The system must allow suppliers to view information about their services and tourist behavior.
	View Administrator Dashboard: The system must allow administrators to view information about behavior within the platform.
Geolocation and Recommendations	**Locating Businesses on a Map:** The system must have an interactive map that allows tourists to locate businesses.
	Receiving Recommendations Based on Preferences: The system must provide recommendations to tourists based on their interests.

© 2026 ITMA. All rights reserved.

Module	Description
Service Management and Visualization	**Managing services:** The system must allow administrators and providers to add, read, edit, and delete services.
	Managing Tourists: The system must allow administrators to add, read, edit, and delete tourists.
	Managing Suppliers: The system must allow administrators to add, read, edit, and delete suppliers.
	Service Reading: The system must allow tourists to explore services and view additional information about each one.

Table 3. Non-functional requirements summary

Quality Attribute	Description
Scalability	The system must be capable of managing a progressive increase in the number of users, data volume, and frequency of interactions without degrading its performance. It should allow the incorporation of new modules and facilitate its expansion into other tourism niches both within and outside of Costa Rica.
Usability	The system must provide a user-friendly and easy-to-navigate interface for users with different levels of technical skills. Interaction should not require prior training, and the dashboards must offer a clean and clear visual design with organized and straightforward information.
Performance	The system must respond efficiently to user requests, maintaining low response times, especially in critical operations such as service registration, metric queries, and event queue processing.
Reliability	The system must ensure continuous and stable operation, guaranteeing data integrity even in the presence of errors or partial failures. Information must be recoverable without loss after restarts or interruptions.
Security	The system must protect personal and sensitive data through the use of secure authentication, role-based access control, and encrypted storage where necessary. Unauthorized access to information must be prevented.
Maintainability	The system must facilitate the incorporation of new features, fixes, or improvements through a clear and well-documented structure. The architecture must allow component updates without interrupting overall operation or compromising system stability.
Availability	The system must remain constantly operational, with backup, monitoring, and fault-tolerance mechanisms that ensure service continuity with minimal interruptions.

© 2026 ITMA. All rights reserved.

4.2 Architectural overview

Figure 1 illustrates the overall organization of the system from a high-level architectural perspective. It provides details on the interactions between system actors, the core microservices, the messaging infrastructure, the intermediate processing layer, and the data analysis component. To ensure a modular, scalable, and flexible system, aligned with the principles of evolutionary design, the architecture follows a microservices-based approach. This choice allows each functionality or domain of the system to be developed, deployed, maintained, and scaled independently, encouraging progressive growth and enabling adaptation to new needs within the tourism ecosystem. Each microservice encapsulates a specific responsibility (such as user management, provider services, or visitor preferences) and operates with its own independent database to avoid duplication. SQLite was selected for the prototype due to its simplicity and portability, with the option to migrate to PostgreSQL or MongoDB in later stages. Asynchronous communication is enabled through RabbitMQ under the AMQP protocol, which acts as an intermediary between the microservices and the analytics module. Events are published to the queue and processed by a Python-based consumer, which transforms and stores them in a centralized MySQL database optimized for queries and analytics.

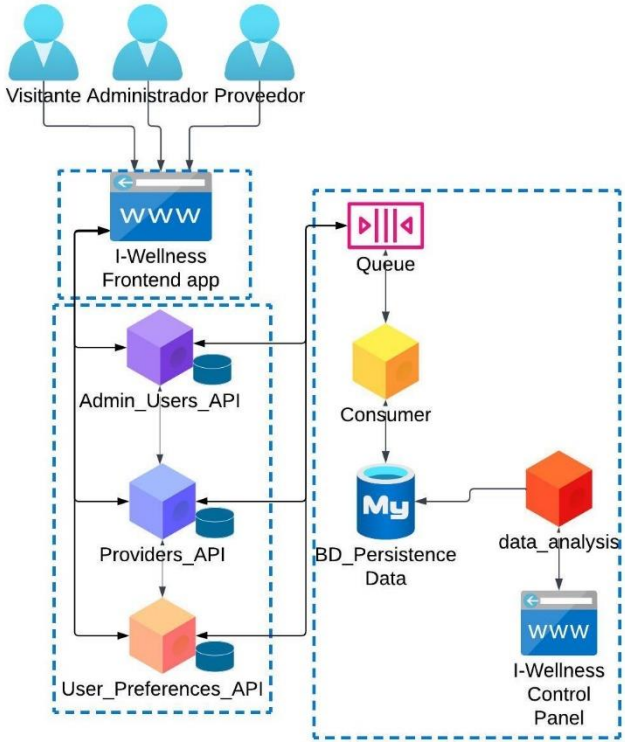


Figure 1. High Level Diagram

The Data_Analysis component, developed in Python with Flask, accesses this consolidated database to generate metrics, reports, and indicators displayed in an interactive web dashboard. This dashboard provides administrators and providers with insights into service demand, user preferences, and territorial indicators. Despite the additional complexity

© 2026 ITMA. All rights reserved.

of managing queues, multiple databases, and different programming languages, the architecture enables horizontal scalability, resilience, technological flexibility, and the future integration of recommendation systems, demand forecasting modules, or artificial intelligence.

Two primitives from Table 1 are already visible at this level. The private store per microservice is what preserves autonomy, since no service can reach another's operational data; the event path through the broker into the consolidated store is what produces transparency, since it yields aggregates queryable by actors who do not own the originating service.

4.3 System context

The context diagram, in Figure 2, represents the main actors and external systems that interact with the service. The visiting user can manage their preferences and consult the different services. The provider user can manage his account, as well as the services belonging to him, in the same way he can filter and manage the metrics in the information dashboard. On the other hand, the administrator user handles all the management of visitor, supplier and service information.

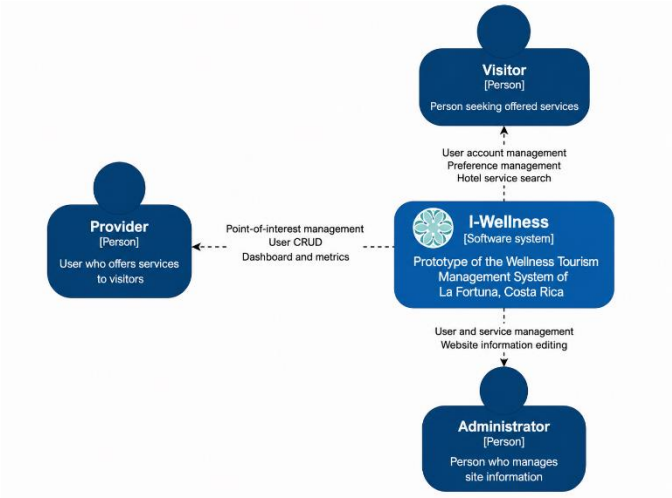


Figure 2. Context Diagram

The governance reading of this level concerns what the system boundary excludes as much as what it contains. The three actor roles correspond to the three participation modes the governance model must sustain: contribution of preference data, provision of services, and stewardship of the shared information base. Notably, the destination management organization is not modelled as a privileged actor with direct access to operational data; it is served through the administrator role and through the same aggregated projections available to providers. This prevents the platform from reproducing, at the level of information access, the asymmetry between institutional and small private actors that characterizes many tourism destinations [19]. The boundary also makes explicit that the system does not integrate with

© 2026 ITMA. All rights reserved.

external booking, payment or public-sector systems in the current version, which leaves fiscal transparency and regulatory reporting outside scope and defines the principal integration surface for subsequent iterations.

4.4 Container view

This diagram, as seen in Figure 3, takes an approach to the system, presenting how it is organized internally. It has a web application developed in TypeScript using the Angular framework, a backend composed of microservices in Java with the Spring Boot framework, and SQLite databases associated with each microservice, where the corresponding data is stored. Additionally, a data persistence and analysis container is included, where everything related to the data for the different information boards will be managed, using technologies such as Python, MySQL, Flask and RabbitMQTT.

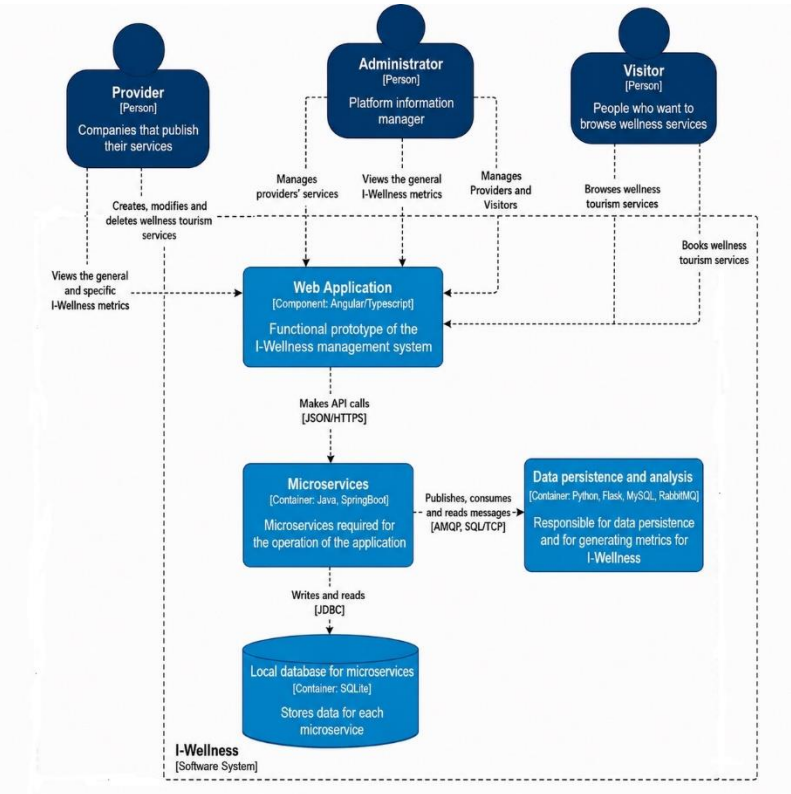


Figure 3. Container Diagram

The governance-relevant property of this decomposition is the strict separation between operational and analytical persistence. Operational stores are private to their owning service and are never queried across boundaries; the analytical store is shared but derived, containing only projections explicitly published as events. This is what allows the platform to offer ecosystem-wide visibility without granting ecosystem-wide access: a provider consulting demand indicators reads the derived store, not the operational records of other providers. The mechanism is architectural rather than procedural — no access-control policy could be

© 2026 ITMA. All rights reserved.

misconfigured in a way that would expose one provider's operational data to another, because no path exists between the two stores. Governance properties enforced by structure are more robust than those enforced by configuration, and this is the principal argument for accepting the operational overhead the decomposition entails.

4.5 Component view

The component diagram in Figure 4 represents the system's logical architecture, structured around three main containers: microservices, microservices databases, and data persistence and analysis. Within the microservices container, three main components developed with Java and Spring Boot are identified: the Provider Services API, responsible for managing the services offered by providers; the User Management API, which handles user registration, authentication, and roles; and the Preferences API, which manages user preferences related to tourism services. Each of these microservices connects to its own database, forming the microservices database container, which consists of three independent SQL database instances: the Services Database, the Users Database, and the Preferences Database. The data persistence and analysis container receives information from the microservices via a RabbitMQ message queue, which acts as an asynchronous communication mechanism. The information is consumed by a persistence component developed in Python, which transforms the messages and stores them in a long-term database (Wellness-long-persistence-db, based on MySQL). Finally, the stored data is used by a dashboard implemented with Flask (Python) and Angular, which allows for the visualization of consolidated information for analysis and decision-making.

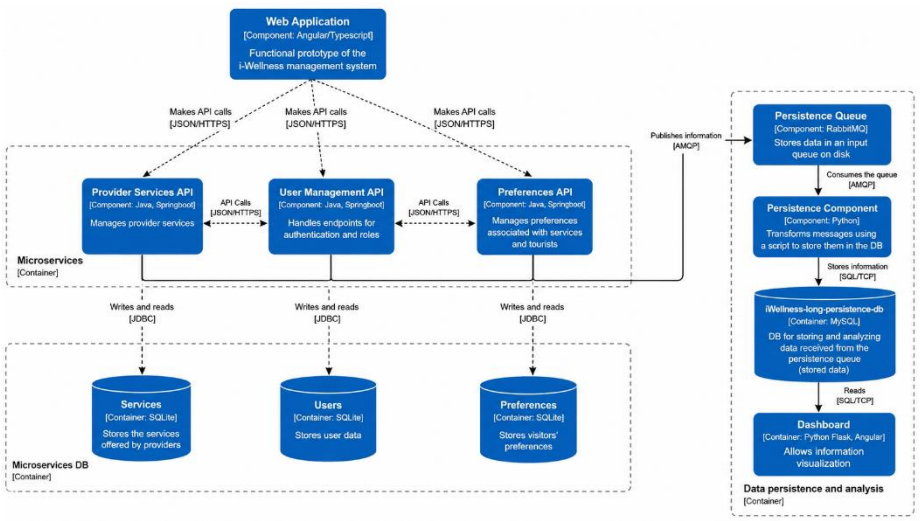


Figure 4. Component diagram

The three microservices are aligned with capabilities rather than with technical layers, which is the decomposition criterion prescribed for microservice systems [23]. This alignment matters for evolvability: the units of change coincide with the units of governance concern, so extending the platform to a new actor category requires adding a bounded context rather than modifying existing ones. Elevating preferences to an independent component rather than

© 2026 ITMA. All rights reserved.

embedding them in the user record is the clearest instance of this criterion at work, and follows directly from the participation principle in Table 1 — a tourist modelled as a co-creator of experiences requires a capability boundary of their own, not an attribute of an account.

4.6 Design rationale: information flow and trade-offs

The preceding views describe the structure of the artifact. This subsection makes explicit the reasoning behind it, since architectural documentation that reports only the option adopted conceals what makes a design assessable.

Two properties of the resulting information flow deserve emphasis. The transition from operational to analytical representation occurs exactly once, in a single component, so the logic determining what the ecosystem can observe is concentrated rather than distributed and is therefore auditable: an actor disputing an indicator can be shown the rule that produced it. And the flow is unidirectional — no path exists from the analytical store back into operational state, so aggregate information can inform actors' decisions but cannot automatically alter another actor's records. Coordination is enabled; unilateral action upon another actor is structurally excluded. This asymmetry distinguishes a governance platform from an orchestration system, and it was an explicit design commitment rather than a consequence of the technologies selected.

Table 4 consolidates the principal design decisions, stating for each the alternatives considered and rejected, the criterion that discriminated between them, the cost accepted, and the governance principle from Table 1 that the decision serves. Decisions are reported as taken during the project, including those that would be revised in a production deployment.

Table 4. Design decisions, alternatives and trade-offs

Decision	Alternatives considered	Rationale	Trade-off accepted	Principle served
Capability-aligned microservice decomposition	Layered monolith; modular monolith	Units of change must coincide with units of governance concern to permit incremental incorporation of actors	Operational complexity: multiple deployment units, distributed debugging, no cross-context transactions	Autonomy; evolvability
One private store per service	Shared relational schema with per-actor views	A shared schema makes autonomy a matter of access-control configuration rather than of structure	Data duplication; eventual rather than immediate consistency across contexts	Autonomy; collaboration

© 2026 ITMA. All rights reserved.

SQLite as operational store	PostgreSQL; MongoDB	Portability and zero administration during a time-bounded project	Constrained concurrent write throughput; migration required before production deployment	— (pragmatic constraint)
Asynchronous AMQP propagation	Synchronous REST aggregation; scheduled batch extraction	Decouples analytical availability from operational responsiveness and buffers information during downstream unavailability	Additional infrastructure component; visibility latency; broker as a coordination dependency	Transparency; evolvability
Separate consolidated analytical store	Querying operational stores directly; federated query layer	Enables ecosystem-wide visibility without ecosystem-wide access	Storage duplication; transformation logic must be maintained	Transparency reconciled with autonomy
Preferences as an independent bounded context	Preference attributes on the user record	Tourists are modelled as co-creators, which requires a capability boundary rather than an attribute	An additional service and store for a comparatively small domain	Participation / co-creation
Unidirectional analytical path	Bidirectional integration permitting automated action on operational state	Coordination should inform decisions, not substitute for them	Forgoes automated cross-actor optimization	Autonomy; accountability
Role-scoped credentials issued by a dedicated service	Per-service local authentication	Attribution of actions to roles must be uniform across contexts	The identity service becomes a single point of failure	Accountability

4.7 Evaluation

The architecture was evaluated through a combination of architectural analysis and technical testing. The Architecture Tradeoff Analysis Method (ATAM) was selected for the former because it assesses an architecture against prioritized quality-attribute scenarios and, unlike checklist-based reviews, explicitly surfaces the points at which quality attributes trade off against one another — which is the concern raised in Section 2.6, where transparency and autonomy were shown to be reconciled by an architectural choice rather than satisfied independently.

The evaluation of the system architecture was carried out through a validation process using the Architecture Tradeoff Analysis Method (ATAM). This validation involved a detailed analysis of the system's functional and non-functional requirements, identifying key

© 2026 ITMA. All rights reserved.

stakeholders and their needs. The provided architectural diagrams, including component, deployment, and data diagrams, were assessed to determine how the architecture meets the requirements and manages various quality attributes. The ATAM process concluded that the proposed architecture demonstrates significant strengths. First, the adoption of microservices provides greater flexibility and scalability, as each system component can be developed, deployed, and scaled independently.

The analysis also identified the sensitivity points, trade-offs and residual risks that the architecture carries, which are reported here because an evaluation that lists only strengths cannot guide subsequent iterations. Two sensitivity points were identified: the throughput of the analytical consumer, on which the visibility latency of every governance indicator depends, and the availability of the User Management API, on which the accountability property of all other services depends. Two trade-off points follow from decisions recorded in Table 4: the propagation mechanism, which sets visibility latency against operational responsiveness, and the granularity of the decomposition, which sets autonomy against operational complexity. Three residual risks remain in the current version — the concentration of identity issuance in a single service without a documented failover path, the write-concurrency ceiling imposed by SQLite at the operational level, and the absence of an implemented replay procedure for reconstructing the analytical store from the event history, which the design makes possible in principle. Conversely, cross-context data exposure was classified as a non-risk, since no path exists between the operational stores of distinct services.

Unit tests were conducted to verify the individual functioning of each system component. The tests covered both the service layer and the controller layer. A 90% code coverage was achieved, executing a total of 90 different test cases, all of which were successfully completed. Integration tests were performed to validate the interaction between system components and functional flows. These tests covered key quality scenarios such as authentication (login and registration for tourists, providers, and administrators), service management (creation, editing, deletion, and status changes), and bookings (creation and cancellation). The results showed a 93% success rate, with 14 of the 15 test cases executed correctly.

Load and stress tests were also conducted to evaluate the system's behavior under realistic usage conditions, with the aim of identifying its maximum capacity, degradation points, and stability during periods of high demand. A total of 558 concurrent users were simulated over a period of three minutes and thirty-eight seconds, calculated to allow the complete execution of all planned scenarios while maintaining precise control over the load variables. Of these, 70% were visitors (approximately 390) and 30% providers.

Two boundaries of this evaluation should be stated explicitly. First, it establishes that the architecture satisfies the quality attributes required for multi-actor coordination, but not that the coordination itself improves: demonstrating the latter would require observing the platform in operation within the destination, which lies beyond the scope of a prototype

© 2026 ITMA. All rights reserved.

assessed under controlled conditions. Second, consistent with the scope defined in Section 3.2, the evaluation reported here is architectural and technical. Participatory validation with ecosystem stakeholders — which would establish whether the capabilities the architecture affords are in fact exercised, and by whom — is carried forward as the first item of the future research agenda in Section 5.

5. Discussion and conclusions

The development of the smart governance prototype for wellness tourism ecosystems offers relevant insights at the intersection of information systems design, digital innovation, and governance models. Beyond its technical scope, the study contributes to the debate on how ICT-based solutions can strengthen coordination, transparency, and shared value creation in complex socio-technical ecosystems. The results suggest that integrating DSR with both structured and agile methodologies is particularly effective in contexts where multiple stakeholders demand rigor and adaptability. This aligns with the perspective of Hevner and Gregor [26], who emphasize the capacity of design science to bridge conceptual models and practical artifacts, while also expanding the discussion of methodological hybridity in IS development highlighted by Pereira et al. [11].

The adoption of a microservices-based architecture also demonstrates that architectural choices have direct implications for governance practices. This finding expands the work of Gajdošík [3] on smart tourism ecosystems and aligns with the principles of platform ecosystems [4], [5] by showing how modular and scalable architectures can concretely support governance principles such as transparency, accountability, and participation. In line with Tomor et al. [6], who argue that socio-technical designs are required to operationalize governance principles, our prototype illustrates how ICT infrastructures can be aligned with collaborative processes in multi-actor ecosystems. The mapping proposed in Section 2.6 sharpens this claim: governance principles are not enacted by digital platforms in general, but by identifiable primitives whose presence leaves observable evidence. Transparency and autonomy, ordinarily in tension, are reconciled by the choice of propagation mechanism rather than by policy — a reconciliation that holds structurally and does not depend on an access-control rule being correctly configured. This reframes the operationalization claim from an assertion into a design criterion that can be specified before construction and audited after it.

Furthermore, this study complements the literature on smart tourism, which has frequently emphasized either the technological dimension of smart destinations [1], [4] or their institutional and organizational challenges [13]. By positioning governance as a mediator between these two spheres, our findings show that effective smart governance depends on their deliberate alignment. The case of La Fortuna, Costa Rica, illustrates how universities, governments, and private actors can work together to design governance solutions tailored to local needs. For destinations composed predominantly of micro and

© 2026 ITMA. All rights reserved.

small enterprises, the practical implication is that architectural investment should be sequenced against governance maturity rather than against technological availability: the transparency and participation primitives are attainable by a small technical team, while autonomy and evolvability become relevant only once multiple institutional actors hold operational data of their own. Ultimately, the findings underscore that smart governance is not merely a technological matter but a socio-technical challenge, in which the sustainable value in tourism emerges from the interplay between governance theory, ICT infrastructures, and collaborative practices.

At the same time, certain limitations must be acknowledged. The mapping between governance principles and architectural primitives establishes necessary conditions for a principle to be attainable in a design; it does not establish that the governance arrangement is effective, equitable, or adopted.

The evaluation focused primarily on technical validation, without extensive involvement of end-users, which limits insights into adoption, usability, and trust. In addition, the project was constrained by time and resources, which restricted the functionalities that could be included. Future research should first undertake the participatory validation delimited in Section 3.2, expand functionalities with AI-driven analytics and personalization, and test the mapping's adaptability across other tourism niches and regions.

Acknowledgments: This research was supported by Pontificia Universidad Javeriana – Bogotá and Instituto Tecnológico de Costa Rica.

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

Declaration of Generative AI Use: The authors used generative AI tools to assist with language editing and formatting consistency. All content was reviewed and verified by the authors.

References

[1] U. Gretzel, M. Sigala, Z. Xiang, y C. Koo, «Smart tourism: foundations and developments», *Electron Markets*, vol. 25, n.º 3, pp. 179-188, sep. 2015, doi: 10.1007/s12525-015-0196-8.

[2] S. Khin y T. C. Ho, «Digital technology, digital capability and organizational performance: A mediating role of digital innovation», *International journal of innovation science*, vol. 11, n.º 2, pp. 177-195, 2019.

[3] T. Gajdošík, *Smart Tourism Destination Governance: Technology and Design-Based Approach*. en Smart Tourism Destination Governance: Technology and Design-Based Approach. 2021, p. 160. doi: 10.4324/9781003269342.

[4] A. Hein *et al.*, «Digital platform ecosystems», *Electron Markets*, vol. 30, n.º 1, pp. 87-98, mar. 2020, doi: 10.1007/s12525-019-00377-4.

© 2026 ITMA. All rights reserved.

[5] M. G. Jacobides, C. Cennamo, y A. Gawer, «Towards a theory of ecosystems», *Strategic Management Journal*, vol. 39, n.º 8, pp. 2255-2276, ago. 2018, doi: 10.1002/smj.2904.

[6] Z. Tomor, A. Meijer, A. Michels, y S. Geertman, «Smart Governance For Sustainable Cities: Findings from a Systematic Literature Review», *Journal of Urban Technology*, vol. 26, n.º 4, pp. 3-27, oct. 2019, doi: 10.1080/10630732.2019.1651178.

[7] L. Errichiello y R. Micera, «A process-based perspective of smart tourism destination governance», *European Journal of Tourism Research*, vol. 29, 2021, doi: 10.54055/ejtr.v29i.2436.

[8] T. Pencarelli, «The digital revolution in the travel and tourism industry», *Inf Technol Tourism*, vol. 22, n.º 3, pp. 455-476, sep. 2020, doi: 10.1007/s40558-019-00160-3.

[9] S. Ivanov, «The impact of automation on tourism and hospitality jobs», *Inf Technol Tourism*, vol. 22, n.º 2, pp. 205-215, jun. 2020, doi: 10.1007/s40558-020-00175-1.

[10] H. J. Scholl y S. AlAwadhi, «Smart governance as key to multi-jurisdictional smart city initiatives: The case of the eCityGov Alliance», *Social Science Information*, vol. 55, n.º 2, pp. 255-277, jun. 2016, doi: 10.1177/0539018416629230.

[11] G. V. Pereira, P. Parycek, E. Falco, y R. Kleinhans, «Smart governance in the context of smart cities: A literature review», *IP*, vol. 23, n.º 2, pp. 143-162, jun. 2018, doi: 10.3233/IP-170067.

[12] A. dos Santos Júnior, L. Mendes Filho, F. Almeida García, y J. Manuel Simões, «Smart Tourism Destinations: un estudio basado en la visión de los stakeholders.», *Revista Turismo em Análise*, vol. 28, n.º 3, 2017, Accedido: 2 de junio de 2025. [En línea]. Disponible en: https://search.ebscohost.com/login.aspx?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=19844867&AN=136411626&h=zn3wluzoyVKrRr6WrGy6tXNHGkopZGfuPoct6gknY7Hk7KDm%2BKDUrbzfAbj7IsGYS2lqfjegyNvliNkh3RHmqA%3D%3D&crl=c

[13] U. Gretzel y M. Scarpino-Johns, «Destination resilience and smart tourism destinations», *Tourism Review International*, vol. 22, n.º 3-4, pp. 263-276, 2018.

[14] G. Antonelli y G. Cappiello, *Smart development in smart communities*. Routledge Oxfordshire, UK, 2017. Accedido: 21 de mayo de 2025. [En línea]. Disponible en: https://api.taylorfrancis.com/content/books/mono/download?identifierName=doi&identifierValue=10.4324/9781315641850&type=googlepdf

[15] R. Kohli y N. P. Melville, «Digital innovation: A review and synthesis», *Information Systems Journal*, vol. 29, n.º 1, pp. 200-223, ene. 2019, doi: 10.1111/isj.12193.

[16] V. Kaptelinin y B. A. Nardi, *Acting with Technology: Activity Theory and Interaction Design*. MIT Press, 2006.

© 2026 ITMA. All rights reserved.

[17] J. Lee y N. Berente, «Digital Innovation and the Division of Innovative Labor: Digital Controls in the Automotive Industry», *Organization Science*, vol. 23, n.º 5, pp. 1428-1447, oct. 2012, doi: 10.1287/orsc.1110.0707.

[18] Y. Yoo, O. Henfridsson, y K. Lyytinen, «Research Commentary—The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research», *Information Systems Research*, vol. 21, n.º 4, pp. 724-735, dic. 2010, doi: 10.1287/isre.1100.0322.

[19] E. Ortiz-Pabón, Lady Fernández-Mora, N. Nova-Arévalo, y R. González-Rivera, «Desafíos Digitales: Análisis de la Madurez Digital de un Clúster Turístico en Seis Cantones de Alajuela, Costa Rica», *Revista Ibérica de Sistemas e Tecnologias de Informação*, n.º E68, pp. 57-74, 2024.

[20] M. R. Kramer, «Creating Shared Value», *Harvard Business Review*, 2011, Accedido: 2 de junio de 2025. [En línea]. Disponible en: http://www.relativimpact.com/downloads/HBR-Shared-value.pdf

[21] E. Ortiz-Pabón, *Ecosistemas de negocios, innovación y emprendimiento: Un marco de referencia y un caso de aplicación*. Editorial Pontificia Universidad Javeriana, 2023.

[22] M. Burkhalter, C. Betz, S. Auge-Dickhut, y R. Jung, «Orchestrating Value Co-Creation in Business Ecosystems», en *Theories of Change: Change Leadership Tools, Models and Applications for Investing in Sustainable Development*, K. Wendt, Ed., Cham: Springer International Publishing, 2021, pp. 257-291. doi: 10.1007/978-3-030-52275-9_16.

[23] I. Nadareishvili, R. Mitra, M. McLarty, y M. Amundsen, «Microservice Architecture». O'Reilly Media, Inc.(8 2016), 2017. Accedido: 28 de agosto de 2026. [En línea]. Disponible en: https://www.corisys.ru/wp-content/uploads/2020/10/microservice-architecture-aligning-principles-practices-and-culture.pdf

[24] E. Ortiz, L. Fernández, N. Nova, y J. Gómez, «From Big Idea to Prototype: A Challenge-Based Learning Framework for Hackathons in Higher Education», en *In Proceedings of the 17th International Conference on Computer Supported Education - Volume 2*, Porto, Portugal, 2025, pp. 670-677. doi: 10.5220/0013233400003932.

[25] M. Treviño-Villalobos, L. Cardinale-Villalobos, L. Fernandez-Mora, L. Víquez-Acuña, y R. González-Quiróz, «Design of an Information System to impulse a process of transformation into a Smart Tourism Destination», *Revista Rosa dos Ventos-Turismo e Hospitalidade*, vol. 14, n.º 3, 2022. Accedido: 7 de julio de 2025. [En línea]. Disponible en: https://scholar.google.com/scholar?hl=es&as_sdt=0%2C5&q=Villalobos%2C+Cardinale%2C+Fern%C3%A1ndez%2C+V%C3%ADquez%2C+Gonz%C3%A1lez%2C+2021&btnG=

[26] A. Hevner y S. Gregor, «Envisioning entrepreneurship and digital innovation through a design science research lens: A matrix approach», *Information & Management*, vol. 59, n.º 3, p. 103350, 2022.

© 2026 ITMA. All rights reserved.