ISBN: 978-628-96613-3-0. ISSN: 2414-6390. Digital Object Identifier: https://dx.doi.org/10.18687/LACCEI2026.1.1.1321

An Engineering Framework for Designing and Evaluating Privacy-by-Design (PbD) in Public Digital Platforms

Roxana Martínez
Universidad Siglo 21, Argentina, roxana-martinez@mi.21.edu.ar

Abstract*– Today, public digital platforms increasingly process personal and sensitive data through complex and interconnected information systems. This necessitates analyzing different engineering approaches that integrate privacy requirements from the earliest stages of system design. While Privacy by Design (PbD) has been analyzed from a regulatory and conceptual perspective, its implementation on public digital platforms remains a significant engineering challenge. This paper proposes an engineering framework for designing and evaluating Privacy by Design on public digital platforms. The proposed framework applies PbD principles to concrete engineering artifacts by structuring privacy integration in three dimensions: a) design requirements, b) technical safeguards, and c) evaluation indicators. These approaches support both decision-making during the design phase and post-implementation evaluation, enabling systematic assessment throughout the platform's lifecycle. To validate the proposed framework, a case-based evaluation was conducted on a set of public digital platforms in the Argentine public sector. An automated prototype was developed to evaluate Privacy by Design (PbD) benchmark indicators, including secure communication mechanisms, transparency features, and observable data minimization practices in user-accessible components. The results reveal recurring engineering deficiencies related to default security configurations, limited visibility of privacy policies, and inadequate handling of sensitive data. This proposal provides a replicable approach that supports engineering practices in the public sector, enabling developers and designers to implement Privacy by Design beyond mere regulatory compliance. This contribution promotes privacy-respecting engineering practices and supports the development of more reliable public digital systems in terms of privacy.

***Keywords:* Privacy-by-Design, Engineering Framework, Public Digital Platforms, Privacy Engineering, Digital Systems Design.**

I. Introduction

Public digital platforms are a central component of public administration, enabling large-scale interactions between governments and citizens through digital services. As various studies point out, "*research on government service quality can help ensure the success of digital government services and has been the focus of numerous studies that proposed different frameworks and approaches. Most of the existing studies are based on traditional researcher-led methods, which struggle to capture the needs of citizens*" [1].

A. Public Digital Platforms and Privacy as an Engineering Challenge

These platforms routinely process personal and sensitive data, such as identification, health, and socioeconomic information, within complex and interconnected information systems. In this context, it is important to understand that "*personal data is a powerful tool. The more someone knows about us, the more power they have over us. But who will control most of our personal data? Does the government and big tech really care about our personal data?*" [2]. Consequently, privacy has shifted from a purely regulatory concern to a crucial engineering challenge that directly impacts system reliability, public trust, and long-term sustainability. In recent years, the concept of PbD has gained traction as a practical approach to integrating data protection principles into information systems from their initial stages. However, several authors caution that while PbD promotes the proactive integration of privacy safeguards throughout the system lifecycle, its effective implementation remains problematic. In this sense, it has been noted that "*Privacy by Design (PbD) is attracting considerable resources and focus, logically encouraging data protection as best practice across the data lifecycle; however, its implementation remains a challenge, as many developers struggle to balance functionality and privacy due to insufficient guidelines and resources*" [3]. Other studies agree that, despite its promising outlook, the engineering implementation of PbD principles is still in its early stages [4]. This has led to scenarios where, although PbD has been widely addressed in regulatory frameworks and conceptual debates, its practical adoption on public digital platforms remains uneven and fragmented from an engineering perspective.

B. Privacy-by-Design from an Engineering Perspective

From an engineering standpoint, the primary challenge lies in translating PbD principles into concrete design requirements, technical controls, and evaluative mechanisms that can be systematically applied and verified. Many existing approaches emphasize compliance or high-level recommendations, offering limited guidance on how privacy can be operationalized through engineering artifacts and processes. This gap is particularly evident in public sector environments, where legacy infrastructures, heterogeneous architectures, and institutional constraints complicate the consistent integration of privacy requirements.

Within the engineering literature, Security-by-Design (SbD) and Privacy-by-Design (PbD) are distinguished as related but conceptually different approaches. While SbD focuses on ensuring system robustness, resilience, and protection against technical threats, PbD specifically addresses how personal data is collected, processed, and managed to protect individuals' privacy rights. This distinction highlights that secure systems are not inherently privacy-respecting, reinforcing the need for engineering frameworks explicitly oriented towards PbD alongside traditional security controls [5]. Consequently, this paper focuses on PbD, treating privacy as a distinct engineering concern and proposing a structured framework to support its integration and systematic evaluation on public digital platforms. The article proposes an engineering framework for the design and evaluation of PbD that structures privacy integration into three complementary dimensions: (a) design requirements, (b) technical safeguards, and (c) evaluation indicators. By aligning these dimensions with requirements engineering, implementation engineering, and validation engineering activities, the framework facilitates both decision-making at the design stage and post-implementation evaluation throughout the platform's lifecycle.

To demonstrate the applicability of the proposed framework, a case-based validation was conducted on a set of public digital platforms within the Argentine public sector. An automated prototype tool was developed to assess observable PbD indicators in user-accessible components, enabling the identification of recurring engineering deficiencies related to security configurations, transparency mechanisms, and the handling of sensitive data.

Accordingly, the main contributions of this paper are summarized as follows:

- The definition of a replicable engineering framework that implements PbD for public digital platforms.
- The alignment of PbD principles with concrete engineering roles, artifacts, and evaluation mechanisms.
- Empirical insights derived from case-based validation that illustrate common shortcomings in privacy engineering and opportunities for improvement.

These contributions aim to support privacy-aware engineering practices and to foster the development of more reliable and trustworthy public digital systems.

II. Background and Related Engineering Work

To improve readability, the discussion in this section has been streamlined to reduce redundancy and emphasize the most relevant engineering perspectives.

Public digital platforms operate within complex socio-technical environments where privacy requirements must be addressed alongside functional, security, and performance constraints. In engineering research [6][7][8], PbD has been increasingly discussed as a fundamental principle for integrating data protection considerations into systems design and development processes [9][10][11]. However, much of the existing literature analyzes PbD from regulatory [12][13], organizational, or conceptual perspectives, offering limited guidance on how to systematically implement privacy principles in engineering workflows. From a software and systems engineering perspective, early efforts to address privacy focused primarily on compliance-oriented checklists or high-level design recommendations. These approaches often lack explicit correspondence between privacy principles and concrete engineering elements, such as requirements specifications, architectural decisions, or validation mechanisms. As a result, privacy considerations are often addressed on an ad hoc basis, depending on the individual experience of the developer using them, rather than being integrated into structured engineering processes.

In parallel, the field of privacy engineering has emerged in response to these limitations, seeking strengthened standardized methods, tools, and techniques for integrating privacy into the design and implementation of systems [14]. Privacy engineering research highlights the need to translate abstract privacy principles into practical requirements, technical safeguards, and measurable indicators [15]. However, existing privacy engineering frameworks often focus on specific application domains or assume homogeneous technological environments, limiting their applicability to public digital platforms characterized by legacy systems, heterogeneous architectures, and institutional constraints. Engineering literature also distinguishes PbD from closely related approaches such as SbD [16]. While security engineering focuses on protecting systems against technical threats through mechanisms such as authentication, authorization, and encryption, privacy engineering addresses broader issues related to data minimization, purpose limitation, transparency, and user control [17][18]. Studies consistently highlight that secure systems do not inherently preserve privacy, highlighting the need for engineering frameworks that explicitly address privacy as a specific design goal, rather than as a byproduct of security controls. Despite the advances, a significant gap persists between the conceptual principles of PbD and its consistent application in public digital platforms. Existing studies rarely provide integrated engineering frameworks that support both the integration of privacy requirements at design time and the post-implementation evaluation of privacy-related properties. This gap underscores the need for engineering-oriented approaches that are replicable and adaptable to real-world public sector environments. The framework proposed in this article addresses this need by structuring PbD into complementary engineering dimensions and supporting its systematic evaluation throughout the system lifecycle.

III. Engineering Framework for Privacy-by-Design

This section presents the proposed Engineering Framework for Privacy by Design (PbD), which aims to support the integration and systematic evaluation of privacy requirements in public digital platforms. The framework proposes a simple, engineering-friendly solution for applying

general PbD principles and their practical implementation in real-world systems, particularly in public sector environments characterized by institutional constraints, heterogeneous architectures, and legacy infrastructures. Rather than addressing privacy as an abstract or compliance-based concern, the proposed framework translates PbD principles into concrete engineering tools that can be applied throughout the system lifecycle. It facilitates decision-making during the design phase and post-implementation evaluation, allowing privacy considerations to be integrated into standard engineering practices, which is highly convenient for its application.

A. Framework Objectives and Design Rationale

The primary objective of the proposed framework is to implement Privacy by Design from an engineering perspective, enabling development teams to integrate privacy considerations as a priority during system design, implementation, and validation. Unlike approaches that address privacy retrospectively or solely through regulatory compliance, the framework promotes proactive and structured integration, aligned with established engineering workflows. The framework is based on three fundamental design assumptions. First, privacy requirements must be explicitly identified and specified during the early stages of system design to avoid ad-hoc or reactive solutions. This also considers that such solutions are very costly and difficult to correct after implementation, which is particularly relevant in government environments that may lack the budget to address such situations. Second, privacy principles must be implemented through observable and verifiable technical safeguards, ensuring that privacy is not only declared but also effectively implemented in system components. Third, Privacy by Design requires evaluation mechanisms to monitor the status of privacy-related properties over time, recognizing that privacy risks evolve alongside systems and usage contexts. To achieve these objectives, the framework adopts a lightweight and modular structure. This design choice facilitates its application across diverse public sector environments, prioritizing replicability and adaptability over prescriptive, domain-specific solutions. By aligning privacy integration with existing engineering functions and artifacts, the framework minimizes adoption barriers and facilitates gradual integration into ongoing development processes.

B. Framework Architecture and Engineering Dimensions

As explained in the previous section, the proposed framework structures the integration of Privacy by Design into three complementary engineering dimensions: (a) Design Requirements, (b) Technical Safeguards, and (c) Evaluation Indicators. Each dimension represents a set of engineering activities and artifacts that collectively enable the integration of privacy throughout the system lifecycle.

Figure 1 presents this general architecture of the engineering framework for integrating Privacy by Design into public digital platforms. The framework structures the incorporation of privacy throughout the system lifecycle using three complementary engineering dimensions: Design Requirements, Technical Safeguards, and Evaluation Indicators. Each of these dimensions is represented by the phases of: a) design, b) development and deployment, and c) operation and review.

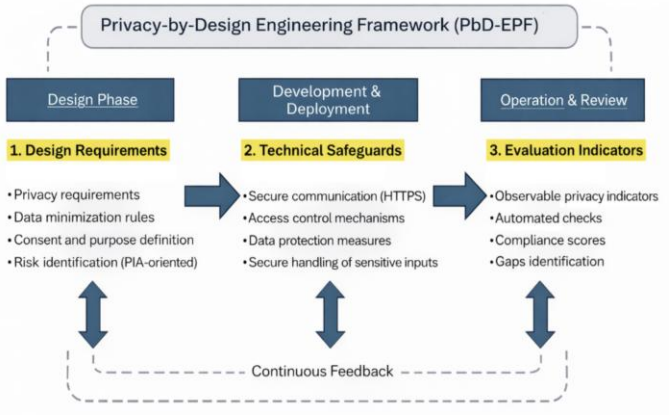


Fig. 1 Privacy-by-Design Engineering Framework (PbD-EPF).

Furthermore, the "Design Requirements" dimension addresses decisions made prior to system construction, including defining privacy requirements, data minimization rules, purpose limitation, consent considerations, and early risk identification for privacy impact assessments. The "Technical Safeguards" dimension focuses on implementing concrete mechanisms that fulfill these requirements, such as secure communications, access controls, data protection measures, and secure handling of sensitive input. The "Evaluation Indicators" dimension provides verification and validation mechanisms through observable privacy indicators, automated checks, and compliance metrics, enabling the identification of potential shortcomings. Finally, the arrows between dimensions represent the flow of engineering activities throughout the system's lifecycle, while the continuous feedback loops reflect the iterative nature of the approach, allowing the evaluation results to inform redesign decisions and continuous improvement. In other words, the framework integrates privacy as a cross-cutting engineering attribute, addressed systematically from initial design to system operation. Each dimension is detailed below:

1) Design Requirements: the Design Requirements dimension focuses on integrating privacy during the early stages of system conception and requirements engineering. Its purpose is to ensure that privacy-related constraints are explicitly identified, specified, and documented before making architectural and implementation decisions. Key elements associated with this dimension include privacy requirement specifications, data minimization rules, purpose limitation definitions, consent considerations, and preliminary information for identifying privacy risks. By formalizing these elements within requirements engineering activities, the framework promotes treating privacy as a fundamental system requirement, rather than an optional or secondary feature. This

dimension facilitates proactive decision-making, enabling development teams to anticipate privacy risks, define acceptable limits for data processing, and establish clear criteria for the use of personal data before system implementation. As a result, subsequent design and implementation activities are guided by explicitly articulated privacy objectives.

2) Technical Safeguards: the Technical Safeguards dimension focuses on implementing privacy requirements through concrete engineering controls and mechanisms. This dimension addresses implementation engineering activities, where abstract privacy principles are translated into operational system features. Typical safeguards include secure communication mechanisms, access control configurations, protection of data in transit and at rest, and secure handling of sensitive inputs within user-facing components. Rather than prescribing specific technologies or tools, the framework emphasizes implementing observable, verifiable safeguards directly traceable to predefined design requirements. By explicitly linking technical safeguards to privacy requirements at design time, the framework improves traceability and accountability, reducing reliance on implicit assumptions or informal practices. This facilitates transparency in the data processing flow for managing that process. Furthermore, this linkage facilitates systematic verification and future system evolution without compromising privacy objectives.

3) Evaluation Indicators: the Evaluation Indicators dimension focuses on validating and evaluating the integration of Privacy by Design throughout the system lifecycle. This dimension supports validation engineering activities by defining measurable indicators, such as metrics, and observables that allow for the assessment of privacy-related properties in implemented systems, leading to better decision-making among potential risk scenarios. Evaluation artifacts include automated assessment checks, compliance and transparency indicators, gap identification reports, and comparative metrics derived from system observations. These indicators support both one-off assessments and continuous monitoring, enabling iterative improvement and informed decision-making. It is important to note that this dimension recognizes that privacy integration is not a one-off activity, but rather an ongoing process that must adapt to system evolution, regulatory changes, and emerging risks. By incorporating evaluation mechanisms, the framework supports sustained Privacy by Design practices beyond the initial implementation. To complement the conceptual description of the framework's dimensions, Table I is used, which presents a structured relationship between each dimension, its corresponding engineering role, and the main elements involved in its implementation. This relationship shows how the principles of Privacy by Design are implemented through standard engineering activities, clarifying the responsibilities, elements, and objectives associated with each stage of the framework. Through this articulation, the engineering proposal of the framework and its practical adoption on public digital platforms are presented.

TABLE I
MAPPING OF FRAMEWORK DIMENSIONS

Framework Dimension	Engineering Role	Primary Artifacts	Purpose within PbD Integration
Design Requirements	Requirements Engineering	Privacy requirements specifications, data minimization rules, consent definitions, risk identification inputs (PIA-oriented)	Define privacy constraints and requirements at early design stages, ensuring that data protection principles are embedded before system implementation
Technical Safeguards	Implementation Engineering	Secure communication mechanisms (HTTPS), access control mechanisms, data protection measures and secure handling of sensitive inputs	Operationalize privacy requirements through concrete technical controls and implementation decisions within public digital platforms
Evaluation Indicators	Validation Engineering	Observable privacy indicators, automated assessment checks, compliance scores, gap identification reports	Enable systematic evaluation and continuous monitoring of Privacy-by-Design integration throughout the system lifecycle

C. Engineering Roles and Lifecycle Integration

Each dimension of the framework aligns with a specific engineering role within the system lifecycle, as shown in Figure 1 and Table I. It is important to note that the Design Requirements dimension corresponds to requirements engineering activities, Technical Safeguards align with implementation engineering, and Evaluation Indicators support validation engineering. This alignment facilitates adoption by multidisciplinary teams and allows integration into existing development workflows without requiring significant organizational restructuring. By working with the design, implementation, and evaluation stages, the framework enables privacy to be addressed by design in a consistent manner throughout the entire lifecycle of public digital platforms. The feedback mechanisms illustrated in Figure 1 facilitate continuous refinement, allowing information derived from evaluation activities to inform future design decisions and iterative system improvements.

IV. CASE-BASED VALIDATION AND PROTOTYPE TOOL

This section presents the case-based validation of the proposed engineering framework for Privacy by Design, as well as the prototype tool developed to support its application. The objective of this validation is not to assess regulatory compliance, but rather to analyse the framework's ability to systematically identify observable privacy properties from an engineering perspective on real-world public digital platforms.

A. Context of the Case Study

The validation considered a set of twelve open-access public digital platforms belonging to the Argentine public sector, selected to represent common categories of citizen-oriented services and heterogeneous technological characteristics. The platforms analyzed include identity management systems, health information services, social assistance platforms, administrative portals, and citizen request reception systems. These platforms provide citizen-oriented digital services and process personal data and, in some cases, sensitive data through web interfaces. For the selection of cases, priority was given to publicly accessible platforms with active use by citizens and representative of common service categories in digital government environments. This context is suitable for evaluating the applicability of the proposed framework under realistic engineering constraints. Although the selection aimed to cover a representative range of public digital services, it is important to acknowledge that the sample may introduce a degree of bias toward web-based, citizen-facing platforms with publicly accessible interfaces. Platforms with restricted access, backend-oriented architectures, or non-web interfaces were not included, which may limit the generalizability of the results across all types of public digital infrastructures. Nevertheless, the selected cases provide a relevant baseline for evaluating observable Privacy-by-Design practices in real-world scenarios.

B. Evaluation Methodology

The evaluation methodology was designed in direct alignment with the Evaluation Indicators dimension of the proposed framework. A case-based approach was adopted, focused on observing technical and interface properties that can be systematically evaluated without privileged access to internal documentation or source code. The evaluation process was structured in three main stages. First, a set of observable Privacy by Design indicators was defined, derived from the framework's dimensions and associated engineering artifacts. Second, an automated prototype tool was developed to analyse these indicators on the selected platforms. Finally, the results were aggregated and analyzed to identify recurring patterns from an engineering perspective. This methodology prioritizes replicability and scalability, allowing the framework to be applied consistently across multiple platforms using homogeneous evaluation criteria. To strengthen the credibility of the evaluation, a complementary manual audit was conducted on a small subset of the analyzed platforms. This audit focused on validating the accuracy of selected heuristic indicators, particularly those related to the detection of privacy policies and the identification of sensitive data fields in web forms. The results of this cross-checking process indicate that the automated heuristics provide a reasonable approximation of observable privacy-related elements, although some discrepancies were identified in cases involving non-standard terminology or implicit policy references. These findings reinforce the suitability of the approach for large-scale comparative analysis while acknowledging inherent limitations of heuristic-based detection.

C. Description of the Prototype Tool

To support the evaluation process, an automated prototype tool was developed to analyse Privacy by Design indicators in components accessible to users of public digital platforms. The tool performs a series of automated checks focused on technical properties and observable indicators that reflect privacy-related engineering decisions. The prototype evaluates, among other aspects, the presence of secure communication mechanisms, the visibility and accessibility of privacy policies, the use of cookies and tracking technologies, and the handling of sensitive data in web forms.

1) Functional scope and prototype requirements: the prototype tool was designed as an automated evaluation mechanism, aligned with the Evaluation Indicators dimension of the proposed engineering framework. Its functional scope is intentionally limited to the analysis of externally observable properties on public digital platforms, enabling systematic evaluation without requiring access to internal documentation, source code, or privileged credentials. The selection of functional requirements (FRs) was aligned with the proposed Privacy by Design framework using observable, replicable, and engineering-oriented indicators. Rather than seeking exhaustive coverage of all possible privacy requirements, the FRs were defined as a minimal but representative set of evaluable properties, aligned with the framework's three dimensions: Design Requirements, Technical Safeguards, and Evaluation Indicators. Each FR was selected based on three criteria. First, the requirement had to be externally observable from user-accessible components, allowing for evaluation without access to internal documentation, source code, or privileged system credentials. Second, the requirement had to reflect concrete engineering decisions, allowing for the evaluation of Privacy by Design principles through technical artifacts rather than abstract statements. Third, the requirements, taken together, had to cover the different stages of the system lifecycle, supporting both design considerations and post-implementation evaluation. This selection approach is consistent with the privacy engineering literature, which emphasizes translating high-level privacy principles into practical and verifiable engineering indicators. Furthermore, the defined RFs do not constitute a comprehensive privacy assessment, but rather an engineering-oriented baseline suitable for comparative analysis across heterogeneous public digital platforms.

From a functional perspective, the prototype meets the following main requirements. Functional Requirements of the Evaluation Prototype:

Dimension 1 — Design Requirements (DR)

(FR1) Automatically detect the adoption of HTTPS as a basic design requirement, verifying the availability of secure communication from the initial access to the platform. (FR2) Identify and evaluate the accessibility and visibility of privacy

policies or other notices related to the processing of personal data from the user interfaces. (FR3) Analyse the presence of explicit statements regarding the purpose of using personal data, as an indicator of declarative transparency in the early stages of design.

Dimension 2 — Technical Safeguards (TS)

(FR4) Detect the implementation of observable secure communication mechanisms, including the use of TLS or other basic transport-level security configurations. (FR5) Identify the presence of basic access control mechanisms, such as authentication, login, or session management functionalities visible on the platform. (FR6) Detect data collection mechanisms through web forms and identify fields potentially associated with personal or sensitive data, as an indicator of critical input protection.

Dimension 3 — Evaluation Indicators (EI)

(FR7) Analyse the use of cookies and tracking technologies exposed on the client side, identifying the absence or presence of associated explicit privacy notices. (FR8) Detect forms that collect personal data without visible references to privacy policies or information on data processing. (FR9) Evaluate the consistency between visible privacy policies and the observable behaviour of the platform, considering the coherence between statements and implemented technical mechanisms.

Table II summarizes the set of functional requirements considered in this study, organized into design requirements, technical safeguards, and evaluation indicators. These requirements capture key aspects of privacy by design in web systems, ranging from secure communication and access control mechanisms to transparency and consistency between declared policies and observable behaviour. The table also links each requirement to concrete, observable indicators, enabling a systematic evaluation of privacy-related practices in real-world web applications.

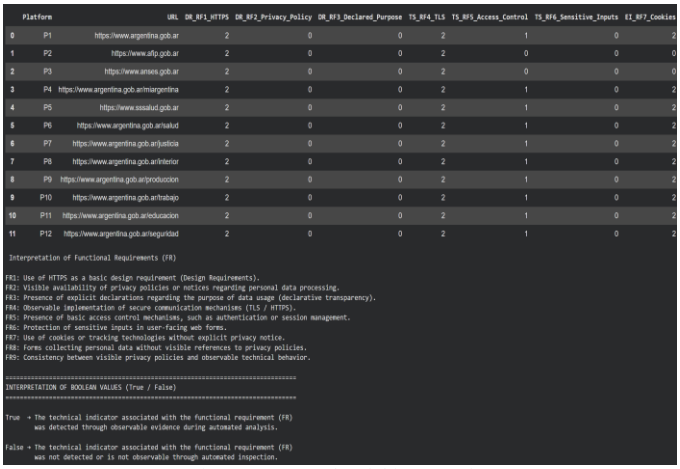
TABLE II
FUNCTIONAL REQUIREMENTS FOR PRIVACY BY DESIGN

Design Requirements	Technical Safeguards	Evaluation Indicators
(FR1) Use of HTTPS as a basic design requirement	(FR4) Implementation of secure communication (observable TLS)	(FR7) Use of cookies without explicit privacy notice
(FR2) Visible availability of privacy policy	(FR5) Presence of basic access control mechanisms (login/session)	(FR8) Forms that collect personal data without reference to privacy
(FR3) Clarity regarding the purpose of data use (declarative transparency)	(FR6) Protection of sensitive inputs in web forms	(FR9) Consistence between visible policies and observable behavior

*2) Technical Implementation of the Prototype: t*he prototype was implemented using the Python programming language, selected for its widespread use in applied research, its large number of libraries for web analytics and data processing, and its ease of replicating experiments in academic settings. The tool was developed as an executable script for notebook environments, facilitating its reuse and adaptation for future extensions of the framework. From a technical standpoint, the prototype follows an external observation-based analysis approach. For each platform evaluated, the tool retrieves publicly accessible HTML content and applies a set of automated checks aligned with the functional requirements defined in the previous section. These checks include analysing secure communication mechanisms, detecting links and content related to privacy policies, identifying web forms and fields associated with personal or sensitive data, and observing tracking technologies exposed on the client side. The prototype execution process is structured as a sequence of repeatable stages: (i) loading the set of URLs to be evaluated, (ii) automated collection of user-accessible content, (iii) evaluation of the indicators associated with each functional requirement, and (iv) generation of structured results. This approach allows for the application of consistent criteria across multiple platforms and facilitates comparative analysis.

As output, the tool generates structured results in tabular format, where each row represents an evaluated platform and each column corresponds to a functional requirement associated with one of the framework's dimensions. These outputs allow for aggregated analysis by dimension, identification of recurring patterns, and detection of implementation aspects from an engineering perspective. To avoid binary interpretations and better reflect partial or incomplete implementations, the evaluation indicators are expressed using a three-level ordinal scale based on observable technical evidence. This scale captures not only the presence or absence of a given Privacy-by-Design indicator, but also intermediate cases where implementation is partial or limited.

Each value is explained below: *0 – Not Detected:* No observable technical evidence associated with the functional requirement was detected through automated analysis; *1 – Partially Detected:* Partial, indirect, or incomplete evidence associated with the functional requirement was detected, indicating limited or fragmented implementation; *2 – Detected:* Clear and observable technical evidence associated with the functional requirement was detected, indicating explicit implementation. An example of this type of output is presented in the Results section, where the indicators obtained are used to analyse common trends in the integration of Privacy by Design. This is shown in Figure 2. On the other hand, Figure 3 illustrates the automated evaluation of web content privacy. The code retrieves web pages with timeout management and analyses their HTML structure to detect the presence of privacy policies and potential references to sensitive data. It searches for privacy-related keywords in links and text content, and scans input fields for indicators of sensitive information, such as credentials, contact details, or health data. This approach provides a rapid method for identifying privacy-relevant elements on websites.



	Platform	URL	DR_RF1_HTTPS	DR_RF2_Privacy_Policy	DR_RF3_Declared_Purpose	TS_RF4_TLS	TS_RF5_Access_Control	TS_RF6_Sensitive_Inputs	EI_RF7_Cookies
0	P1	https://www.argentina.gob.ar	2	0	0	2	1	0	2
1	P2	https://www.afip.gob.ar	2	0	0	2	0	0	0
2	P3	https://www.anses.gob.ar	2	0	0	2	0	0	0
3	P4	https://www.argentina.gob.ar/miargentina	2	0	0	2	1	0	2
4	P5	https://www.ssalud.gob.ar	2	0	0	2	1	0	2
5	P6	https://www.argentina.gob.ar/salud	2	0	0	2	1	0	2
6	P7	https://www.argentina.gob.ar/justicia	2	0	0	2	1	0	2
7	P8	https://www.argentina.gob.ar/interior	2	0	0	2	1	0	2
8	P9	https://www.argentina.gob.ar/produccion	2	0	0	2	1	0	2
9	P10	https://www.argentina.gob.ar/trabajo	2	0	0	2	1	0	2
10	P11	https://www.argentina.gob.ar/educacion	2	0	0	2	1	0	2
11	P12	https://www.argentina.gob.ar/seguridad	2	0	0	2	1	0	2

Interpretation of Functional Requirements (FR)

FR1: Use of HTTPS as a basic design requirement (Design Requirements).
FR2: Visible availability of privacy policies or notices regarding personal data processing.
FR3: Presence of explicit declarations regarding the purpose of data usage (declarative transparency).
FR4: Observable implementation of secure communication mechanisms (TLS / HTTPS).
FR5: Presence of basic access control mechanisms, such as authentication or session management.
FR6: Protection of sensitive inputs in user-facing web forms.
FR7: Use of cookies or tracking technologies without explicit privacy notice.
FR8: Forms collecting personal data without visible references to privacy policies.
FR9: Consistency between visible privacy policies and observable technical behavior.

INTERPRETATION OF BOOLEAN VALUES (True / False)

True → The technical indicator associated with the functional requirement (FR) was detected through observable evidence during automated analysis.

False → The technical indicator associated with the functional requirement (FR) was not detected or is not observable through automated inspection.

Fig. 2 An example of this type of output.

Finally, it is worth noting that the prototype includes an automatic interpretation layer that clarifies the meaning of the Boolean values generated for each functional requirement, reducing ambiguities the replicability of the analysis.

```python
TIMEOUT = 10

PRIVACY_KEYWORDS = ["privacy", "privacidad", "personal data"]
SENSITIVE_INPUTS = ["dni", "cuil", "cuit", "email", "phone", "health", "password"]

# ----------------------------
# Auxiliary functions
# ----------------------------
def fetch_html(url):
    try:
        r = requests.get(url, timeout=TIMEOUT)
        return r.text if r.status_code == 200 else ""
    except:
        return ""

def contains_privacy_policy(soup):
    for link in soup.find_all("a", href=True):
        text = link.get_text().lower()
        href = link["href"].lower()
        if any(k in text or k in href for k in PRIVACY_KEYWORDS):
            return True
    return False

def detect_sensitive_inputs(soup):
    for form in soup.find_all("form"):
        for inp in form.find_all("input"):
            name = (inp.get("name") or "").lower()
            itype = (inp.get("type") or "").lower()
            if any(k in name or k in itype for k in SENSITIVE_INPUTS):
                return True
    return False
```

Fig. 3 Python-based heuristic for automated privacy and sensitive data detection.

It is important to note that the heuristic approach based on keyword matching may introduce both false positive and false negatives. For example, sensitive data fields that use non-standard naming conventions may not be detected, while generic terms may incorrectly be classified as sensitive. Despite these limitations, the approach remains effective for large-scale exploratory analysis, providing a practical trade-off between accuracy and scalability in environments where deeper inspection is not feasible.

D. Evaluated Indicators

The indicators analyzed by the prototype tool were organized into categories aligned with the dimensions of the engineering framework. Technical indicators include the use of secure communication protocols, observable access control configurations at the interface level, and data protection measures applied to the transmission and input of information. Transparency indicators address the availability and clarity of privacy policies and information related to consent. Indicators linked to data minimization and the handling of sensitive fields were also incorporated, as manifestations of design decisions made in early stages. Therefore, these indicators provide a practical basis for evaluating the extent to which Privacy by Design principles are reflected in public digital platforms in production.

E. Scope and Limitations

It is important to note the scope and limitations of the validation approach adopted. The evaluation focuses on externally observable and verifiable indicators and does not analyse internal data processing practices, organizational policies, or backend security controls that are not visible through user-facing components. Consequently, the results should be interpreted as indicators of the quality of the engineering implementation of privacy, and not as comprehensive assessments of legal or organizational compliance. Despite these limitations, the case-based validation demonstrates the viability and usefulness of the proposed framework as an engineering tool for identifying strengths and weaknesses in the integration of Privacy by Design on public digital platforms. As a line of research for future work by the authors, the aim is to incorporate additional indicators, conduct a more in-depth technical inspection, and perform a domain-specific requirements analysis.

V. Results and Discussion

This section presents and analyzes the results obtained from the case-based validation of the proposed Privacy-by-Design Engineering Framework (PbD-EPF). The analysis focuses on observable technical properties derived from the automated evaluation of public digital platforms, with the aim of identifying recurring patterns, strengths, and limitations in the technical integration of the Privacy-by-Design approach.

A. Overall Results by Framework Dimension

Figure 4 presents the average values obtained for each of the framework dimensions: Design Requirements, Technical Safeguards, and Evaluation Indicators. The results show differences between these dimensions. In general, the Design Requirements dimension shows higher average values compared to the Evaluation Indicators dimension, suggesting that basic design decisions, such as the adoption of HTTPS or the availability of visible privacy policies, are more established than the systematic mechanisms for evaluating and verifying consistency between privacy statements and observable technical behaviour. The Technical Safeguards show intermediate values, reflecting a partial implementation of specific protection mechanisms, such as basic access controls or the handling of sensitive data inputs.

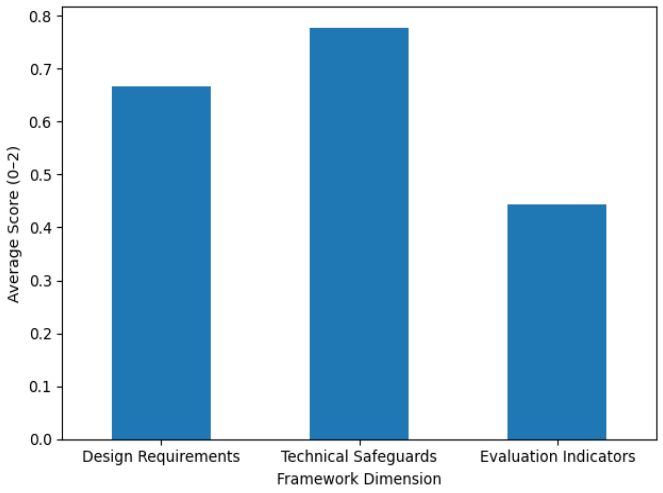


Fig. 4 Average Privacy-by-Design score per framework dimension.

This result highlights the gap between high-level design decisions and their consistent implementation in technical controls, reinforcing the need for engineering frameworks that facilitate traceability between requirements and implementation mechanisms.

B. Distribution of Evaluation Scores

Figure 5 shows the distribution of values obtained on the ordinal scale (0–1–2) for each dimension of the framework. This analysis allows for a more detailed examination of the nature of Privacy-by-Design integration in the evaluated platforms. A significant proportion of the indicators are concentrated at the "partially detected" level, especially in the Technical Safeguards and Evaluation Indicators dimensions.

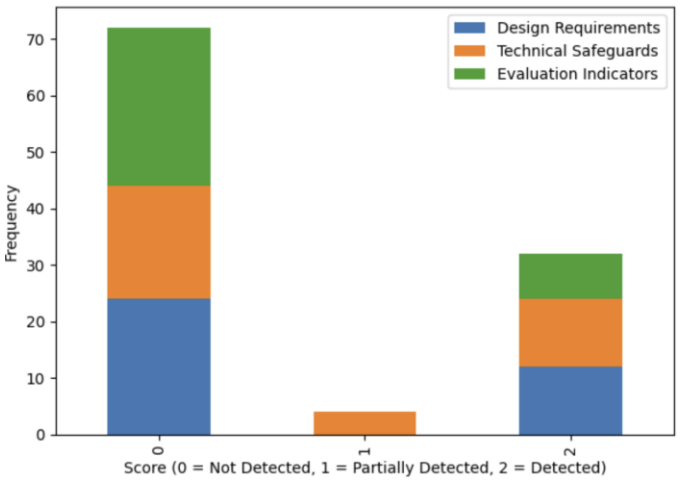


Fig. 5 Distribution of evaluation scores (0–1–2) by framework dimension.

These results indicate fragmented or incomplete implementations, in which certain privacy-related mechanisms are present but lack systematic coverage or clear alignment with stated privacy objectives. The lower frequency of fully detected indicators (value = 2) in the Evaluation Indicators dimension confirms that post-implementation evaluation mechanisms remain limited on the analyzed public digital platforms. The use of an ordinal scale, rather than a binary evaluation, proved suitable for capturing these aspects, allowing for a more realistic representation of engineering practices and avoiding simplistic interpretations of Privacy-by-Design integration. A more detailed inspection of the "partially detected" category reveals recurring implementation gaps across specific functional requirements. For instance, in the case of privacy policy visibility (FR2), partial detection is often associated with policies that are present but difficult to locate or not directly linked to key user interaction points. For sensitive data handling (FR6), partial detection frequently corresponds to forms that request personal information without clear input validation or contextual privacy notices. Similarly, inconsistencies between declared policies and observable behavior (FR9) often result from generic policy statements that are not fully reflected in the platform's technical implementation. These patterns provide actionable insights for engineering teams by highlighting concrete areas where Privacy-by-Design practices tend to degrade in real-world systems.

C. Aggregate Analysis by Platform

Figure 6 presents the aggregate Privacy-by-Design scores by platform, calculated as the average of the functional requirements evaluated in the three dimensions of the framework. The results show considerable variability between platforms, even within the same institutional context.

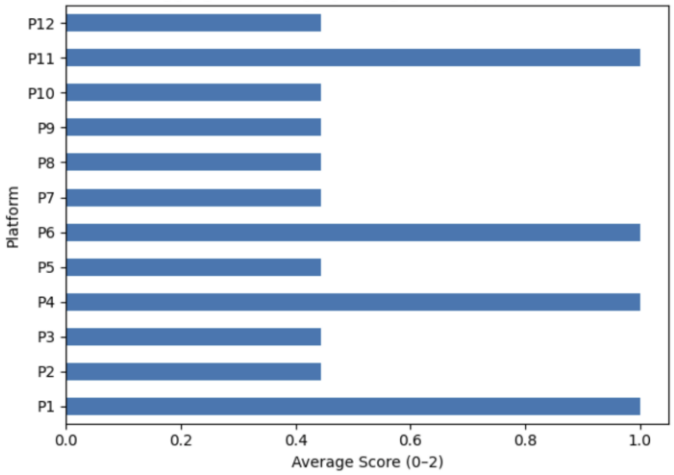


Fig. 6 Aggregate Privacy-by-Design score per platform.

Some platforms achieve relatively higher aggregate scores, reflecting a more consistent integration of privacy-oriented design decisions and technical safeguards. Others, however, exhibit lower scores, generally associated with the absence of visible privacy policies, limited transparency regarding the purpose of data processing, or inconsistencies between privacy statements and observable technical behavior.

This heterogeneity highlights the inherent challenges of the homogeneous integration of Privacy-by-Design in public environments characterized by diverse technological architectures, legacy systems, and decentralized development practices.

D. *Discussion from an Engineering Perspective*

The results confirm the existence of a persistent gap between the conceptual principles of Privacy-by-Design and its effective technical implementation on public digital platforms. While many platforms demonstrate the adoption of basic privacy-related design requirements, fewer systematically incorporate technical safeguards, and even fewer implement explicit evaluation mechanisms to support continuous improvement processes. The proposed PbD-EPF helps bridge this gap by offering a structured approach that links design requirements, technical safeguards, and evaluation indicators within a unified engineering framework. Case-based validation demonstrates that the framework identifies recurring technical deficiencies and provides a replicable basis for comparative analysis of heterogeneous systems. It is important to note that the results should not be interpreted as indicators of regulatory compliance, but rather as evidence of the degree to which privacy considerations are operationalized from an engineering perspective. In this sense, the framework facilitates informed technical decision-making and provides a structured alternative to ad hoc or exclusively compliance-oriented approaches.

VI. Implications for Engineering Practice

The results obtained from case-based validation and the use of the automated prototype allow us to extract a set of relevant implications for the practice of systems and software engineering, particularly in the design, evaluation, and maintenance of digital platforms that process personal data. Unlike approaches focused exclusively on regulatory compliance, the proposed framework is geared toward supporting concrete technical decisions throughout the systems lifecycle.

First, the Engineering Framework for Privacy-by-Design (PbD-EPF) can be used as a support tool during the early design stages, enabling engineering teams to identify technically verifiable privacy requirements. The framework's structuring into dimensions: Design Requirements, Technical Safeguards, and Evaluation Indicators, facilitates the translation of abstract privacy principles into observable design artifacts and technical mechanisms, aligning with privacy engineering approaches that promote the operationalization of privacy through concrete controls [19][20]. Secondly, the framework and prototype developed are especially useful in existing system assessment scenarios, where access to internal documentation, data models, or source code is not always available. The use of external and observable indicators allows for rapid, comparable, and reproducible technical diagnoses, which is particularly relevant in public engineering environments and technical audits of production platforms. This approach reduces reliance on formal declarations or visible policies and focuses on the technical behavior implemented, a limitation widely noted in the literature on Privacy by Design applied to real-world systems [21]. Furthermore, the reproducibility of the approach constitutes a significant practical contribution. The explicit definition of indicators, along with the automation of their evaluation through replicable scripts, allows other engineering teams to apply the framework in different contexts without needing to reinterpret regulatory principles or adapt subjective criteria. In this sense, PbD-EPF can function as an intermediary layer between conceptual privacy frameworks and engineering practices, facilitating homogeneous assessments in heterogeneous systems. From a transferability perspective, although the validation was performed on Argentine public sector platforms, the framework does not incorporate regulatory, institutional, or technological dependencies specific to that context. The selected indicators are based on technical properties common to contemporary web systems, enabling their application in other domains, such as digital health platforms, online education, financial services, or smart governance infrastructures. This regulatory independence reinforces the engineering nature of the approach and expands its potential for adoption in different organizational environments.

Finally, the results obtained provide relevant lessons for public engineering teams. They highlight the need to complement the publication of privacy policies with verifiable technical mechanisms, as well as to incorporate systematic privacy assessments beyond the development stage. The proposed framework can be used as a technical communication tool among development teams, systems administrators, and auditors, contributing to integrating privacy as a quality property of the system and not merely as an external requirement. Taken together, these implications reinforce the value of PbD-EPF as a practical tool for moving from ad-hoc approaches to more systematic, measurable, and reproducible engineering practices in the integration of Privacy by Design into digital systems. In public sector environments, the implementation of Privacy by Design is often constrained by legacy systems, limited budgets, and fragmented governance structures. These conditions make it difficult to introduce systematic privacy controls or redesign existing architecture. As a result, engineering solutions must balance ideal privacy principles with practical constraints, reinforcing the need for lightweight, adaptable, and incremental frameworks such as the one proposed in this work.

VII. Conclusions and Future Work

This work presented an engineering framework for integrating Privacy by Design (PbD) into digital systems, along with its empirical validation through a case study of public platforms in production and the development of a prototype automated evaluation tool. Unlike existing

predominantly normative or conceptual approaches, this proposal focuses on observable technical properties, allowing for the analysis of privacy integration from a systems and software engineering perspective.

The main contribution of this work lies in the operational translation of Privacy by Design principles into a structured set of dimensions, functional requirements, and measurable indicators. The framework's decomposition into Design Requirements, Technical Safeguards, and Evaluation Indicators facilitates the identification of relevant design decisions and their verification in real-world systems, helping to bridge the gap between theoretical PbD and its effective technical implementation.

Case-based validation demonstrated that the framework allows for the identification of recurring patterns in privacy implementation on public digital platforms, as well as inconsistencies between visible declarations and observable technical behaviors. The results show that, while certain basic design mechanisms are reasonably integrated, the systematic processes of evaluation and technical consistency exhibit significant weaknesses. These findings reinforce the need to incorporate continuous engineering evaluation approaches, beyond one-off or documentary verifications.

The developed prototype constitutes a practical contribution that complements this proposal by demonstrating the feasibility of automating the evaluation of PbD indicators using external analysis techniques. Its replicable design makes it a suitable tool for comparative diagnostics, technical audits, and exploration analyses in heterogeneous contexts. However, the results generated by the tool should be interpreted as observable technical evidence, and not as assessments of legal or regulatory compliance.

This study has some limitations. In particular, the evaluation is restricted to indicators visible from the interface and client-side behavior, without considering internal processes, backend data flows, or organizational practices. Furthermore, the set of indicators analyzed is not intended to be exhaustive, but rather representative of an engineering baseline for comparative analysis.

As future lines of work, it is proposed to expand the framework by incorporating new technical indicators, deepen the analysis through dynamic inspections and more advanced tests, and integrate interactive results visualization mechanisms. Overall, the results obtained confirm the potential of the proposed framework as a systematic, reproducible, and transferable engineering tool for advancing a more effective integration of Privacy by Design in complex digital systems.

REFERENCES

[1] Ye, X., Su, X., Yao, Z., Dong, L. A., Lin, Q., & Yu, S. (2023). How do citizens view digital government services? study on digital government service quality based on citizen feedback. *Mathematics*, *11*(14), 3122.

[2] Iman, N. (2024). The fight for our personal data: analyzing the economics of data and privacy on digital platforms. *International Journal of Law and Management*, *66*(6), 774-791.

[3] Obiokafor, I. N., Ajonuma, M. E., & Aguboshim, F. C. (2025). Integrating Privacy by Design (PbD) in the system development life cycle for enhanced data protection. *World Journal of Advanced Research and Reviews*, *26*(01), 1233-1240.

[4] Das, H., Renteria-Pinon, M., Gong, N., & Liu, J. (2025, August). Privacy by Design via Novel Memory Technologies. In 2025 IEEE International Conference on Omni-layer Intelligent Systems (COINS) (pp. 1-7). IEEE.

[5] Del-Real, C., De Busser, E., & van den Berg, B. (2024). Shielding software systems: A comparison of security by design and privacy by design based on a systematic literature review. *Computer Law & Security Review*, *52*, 105933.

[6] Mashaly, B., Selim, S., Yousef, A. H., & Fouad, K. M. (2022, May). Privacy by design: a microservices-based software architecture approach. In *2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)* (pp. 357-364). IEEE.

[7] Kosenkov, O., Zabardast, E., Fucci, D., Mendez, D., & Unterkalmsteiner, M. (2025). Privacy by design: Aligning GDPR and software engineering specifications with a requirements engineering approach. *Information and Software Technology*, 107946.

[8] Kosenkov, O., Zabardast, E., Fischbach, J., Gorschek, T., & Mendez, D. (2026). Towards a Goal-Centric Assessment of Requirements Engineering Methods for Privacy by Design. *arXiv preprint arXiv:2601.16080*.

[9] Duggineni, S. (2023). Impact of controls on data integrity and information systems. *Science and technology*, *13*(2), 29-35.

[10] Li, Y., Wang, R., Li, Y., Zhang, M., & Long, C. (2023). Wind power forecasting considering data privacy protection: A federated deep reinforcement learning approach. *Applied Energy*, *329*, 120291.

[11] Quach, S., Thaichon, P., Martin, K. D., Weaven, S., & Palmatier, R. W. (2022). Digital technologies: tensions in privacy and data. *Journal of the academy of marketing science*, *50*(6), 1299-1323.

[12] Tunca, M. (2025). European and British Regulatory Perspectives: A Systematic Literature Review on Privacy by Design. In *International Conference on Risks and Security of Internet and Systems* (pp. 261-276). Springer, Cham.

[13] Obiokafor, I. N., Ajonuma, M. E., & Aguboshim, F. C. (2025). Integrating Privacy by Design (PbD) in the system development life cycle for enhanced data protection. *World Journal of Advanced Research and Reviews*, *26*(01), 1233-1240.

[14] Andrade, V. C., Gomes, R. D., Reinehr, S., Freitas, C. O. D. A., & Malucelli, A. (2022, November). Privacy by design and software engineering: A systematic literature review. In *Proceedings of the XXI Brazilian Symposium on Software Quality* (pp. 1-10).

[15] Jain, S. (2024). Integrating Privacy by Design Enhancing Cyber Security Practices in Software Development. *Journal Of Multidisciplinary*, *4*(11), 1-11.

[16] Trump, B. D., Antunes, D., Palma-Oliveira, J., Nelson, A., Hudecova, A. M., Rundén-Pran, E., ... & Linkov, I. (2024). Safety-by-design and engineered nanomaterials: the need to move from theory to practice. *Environment Systems and Decisions*, *44*(1), 177-188.

[17] Ben Jeddi, H., Goede, H., Franken, R., van Someren, E., Shandilya, N., Vermoolen, R., ... & Fransman, W. (2025). Development of a nano-specific safe-by-design module to identify risk management strategies. *Annals of Work Exposures and Health*, *69*(3), 310-322.

[18] Sudheshwar, A., Apel, C., Kümmerer, K., Wang, Z., Soeteman-Hernández, L. G., Valsami-Jones, E., ... & Nowack, B. (2024). Learning from Safe-by-Design for Safe-and-Sustainable-by-Design: Mapping the current landscape of Safe-by-Design reviews, case studies, and frameworks. *Environment International*, *183*, 108305.

[19] Kostova, B., Gürses, S., & Troncoso, C. (2020). Privacy engineering meets software engineering. on the challenges of engineering privacy bydesign. *arXiv preprint arXiv:2007.08613*.

[20] Gürses, S., Troncoso, C., & Diaz, C. (2011). Engineering privacy by design. *Computers, Privacy & Data Protection*, *14*(3), 25.

[21] Spiekermann, S. (2012). The challenges of privacy by design. *Communications of the ACM*, *55*(7), 38-40.