**Mapping of *Privacy by Design* Principles to Documentation**

Table 2.1 provides a mapping between the seven PbD principles and the documentation that software engineers SHALL produce or reference throughout the software development lifecycle from software conception to retirement. A checklist column may be added to the table.

Please note spreadsheets, modeling languages, and other tools or representations may be used on their own or in combination for documentation, as long as they are sufficiently powerful to capture the essence of the software engineering translation of the PbD principles as provided in Table 2.1.

An “Annex to the Privacy by Design Documentation for Software Engineers Version 1.0” [PbD-SE-Annex-1.0] accompanies this specification and provides further information, process thinking, explanation of techniques and pedagogical material. It is intended to be helpful to software organizations and engineers implementing the specification.

Table 2.1. Mapping of *Privacy by Design* Principles to Software Engineering Referenced and Generated Documentation

|  |  |  |
| --- | --- | --- |
| **PbD Principle** | **PbD Sub-Principle** | **Documentation** |
| 1. Proactive not Reactive; Preventative not Remedial | **1.1–Demonstrable Leadership**: A clear commitment, at the highest levels, to prescribe and enforce high standards of privacy protection, generally higher than prevailing legal requirements.**1.2–Defined Community of Practice**: Demonstrable privacy commitment shared by organization members, user/data subject communities and relevant stakeholders.**1.3–Proactive and Iterative**: Continuous processes to identify privacy and data protection risks arising from poor designs, practices and outcomes, and to mitigate unintended or negative impacts in proactive and systematic ways. | **SHALL** normatively reference the PbD-SE specification**SHALL** reference assignment of responsibility, accountability and funding for privacy in the organization, and privacy training program(s).**SHALL** include a requirement for employees and service providers to be regularly trained and acknowledge and agree to adhere to privacy policies and notices**SHALL** include the measurement of completeness and effectiveness of privacy training**SHALL** include assignment of resources and job descriptions to the software project, recording who are responsible, accountable, consulted, or informed for various privacy-related tasks**SHALL** reference all external and internal sources of privacy requirements, including regulations strategies, principles, policies, notices and guidelines **SHALL** include privacy requirements specific to the service/product being engineered, and anticipated internal and external deployment environments**SHALL** include privacy risk/threat model(s) including analysis and risk identification, risk prioritization, and controls clearly mapped to risks**SHALL** regularly demonstrate accountability to designated stakeholders  |
| 2. Privacy by Default | **2.1–Purpose Specificity:** Purposes must be specific and limited, and be amenable to engineering controls**2.2–Adherence to Purposes:** Methods must be in place to ensure that personal data is collected, used and disclosed:in conformity with specific, limited purposes;in agreement with data subject consent; andin compliance with applicable laws and regulations**2.3–Engineering Controls:**Strict limits should be placed on each phase of data processing lifecycle engaged by the software under development, including:Limiting Collection;Collecting by Fair and Lawful Means;Collecting from Third Parties;Limiting Uses and Disclosures;Limiting Retention;Disposal, Destruction; and Redaction | **SHALL** imbed privacy into all product, systems and operational development and maintenance processes **SHALL** imbed privacy into all internal and external contracts and legal agreements**SHALL** list all [categories of] data subjects as a stakeholder**SHALL** clearly record the purposes for collection and processing, including the source, collection, storage, use, sharing, trans-border flows securing and retirment of personal data**SHALL** document expressive models of detailed data flows, processes, and behaviors for use cases or user stories associated with internal software project and all data/process interaction with external platforms, systems, operations, jurisdictions, APIs, and/or imported code.**SHALL** describe selection of privacy controls and privacy services/APIs and where they apply to privacy functional requirements and risks.**SHALL**include software and personal data retirement plan from a privacy viewpoint  |
| 3. Privacy Embedded into Design | **3.1–Holistic and Integrative**: Privacy commitments must be embedded in holistic and integrative ways.**3.2–Systematic and Auditable:** A systematic approach should be adopted that relies upon accepted standards and process frameworks, and is amenable to external review.**3.3–Review and Assess:** Detailed privacy impact and risk assessments should be used as a basis for design decisions.**3.4–Human-Proof:** The privacy risks should be demonstrably minimized and not increase through operation, misconfiguration, or error. | **SHALL** use the OASIS PbD-SE Privacy Use Template (see PbD-SE Annex Section 5 [PbD-SE-Annex-1.0]) or the more comprehensive OASIS PMRM methodology [PMRM 1.0] or equivalent for identifying and documenting privacy requirements and developing privacy controls and services**SHALL** contain description of its business model showing traceability of personal data flows for any data collected through new software services under development.**SHALL** include identification of privacy design requirements, controls and services**SHALL** contain a privacy architecture that is integrated into the overall IT and business process architectures**SHALL** describe privacy UI/UX design**SHALL** define privacy and security metrics and accountability measures**SHALL** include human sign-offs/privacy checklists for software engineering artifacts**SHALL** include privacy review reports*(either in reviewed documents or in separate report)*  |
| 4. Full Functionality: Positive Sum, not Zero-Sum | **4.1–No Loss of Functionality:**Embedding privacy adds to the desired functionality of a given technology, process or network architecture.**4.2-Accommodate Legitimate Objectives**: All interests and objectives must be documented, desired functions articulated, metrics agreed, and trade-offs rejected, when engineering software solutions.**4.3–Practical and Demonstrable Results**: Optimized outcomes should be published for others to emulate and become best practices. | **SHALL** treat *privacy-as-a-functional requirement (see section 2.1.4 of the PbD-SE Annex Guide*[PbD-SE-Annex-1.0]*),* i.e. functional software requirements and privacy requirements should be considered together, with no loss of functionality**SHALL**show tests for meeting privacy requirements, in terms of the operation and effectiveness of implemented privacy controls or services  |
| 5. End-to-End Lifecycle Protection | **5.1–Protect Continuously:**Personal data must be continuously protected across the entire domain and throughout the data life-cycle from creation to destruction.**5.2–Control Access:**Controls on access to personal data should be commensurate with its degree of sensitivity, and be consistent with recognized standards and practices.**5.3–Use Security and Privacy Metrics:**Applied security standards must assure the confidentiality, integrity and availability of personal data and be amenable to verification**5.4 Satisfy Privacy Properties:**Wherever possible, software must satisfy properties such as user/data subject comprehension, choice, consent, consciousness, consistency, confinement (setting limits to collection, use, disclosure, retention, purpose), and context(s) around personal data at a functional level; minimized identifiability, linkability, and observability; and maximized traceability, audibility and accountability at a systems level, and be amenable to verification. | **SHALL** be produced for all stages of the software development lifecycle from referencing applicable strategies principles, policies, notices, guidelines and regulations to defining privacy requirements, to design, implementation, maintenance, and retirement**SHALL** reference privacy requirements, risk analyses, controls selection, architectures, design, implementation mechanisms and services, retirement plan, and sign-offs with respect to privacy and security**SHALL** include security and privacy metrics and monitoring practices designed in and/or deployed in the software, or monitoring software, or otherwise in the organization, and across partnering software systems or organizations.**SHALL** demonstrate designs and implementations that satisfy state-of-the-art privacy compliance and innovation  |
| 6. Visibility and Transparency | **6.1–Open Collaboration:**Privacy requirements, risks, implementation methods and outcomes should be documented throughout the development lifecycle and communicated to project members and relevant stakeholders.**6.2–Open to Review:**The design and operation of software systems should demonstrably satisfy the strongest privacy laws, contracts, policies and industry norms (as required).**6.3–Open to Emulation:**The design and operation of privacy-enhanced information technologies and systems should be open to scrutiny, improvement, praise, and emulation by others. | **SHALL***reference*the privacy policies, notices and documentation of all other collaborating stakeholders**SHALL** include description of contextual visibility and transparency mechanisms at the point of contextual interaction with the user/data subject and other stakeholders for data collection, storage, use, sharing, trans-border flows, retirement and/or elsewhere as applicable**SHALL** describe any measurements incorporated in the software, or monitoring software, or otherwise to measure the usage and effectiveness of privacy requirements, options and controls, and to ensure continuous improvement**SHALL** describe placement of privacy settings, privacy controls, privacy notices and accessibility, prominence, clarity, and intended effectiveness |
| 7. Respect for User Privacy | **7.1–Anticipate and Inform:** Software should be designed with user/data subject privacy interests in mind, and convey privacy attributes (where relevant) in a timely, useful, and effective way.**7.2–Support Data Subject Input and Direction:**Technologies, operations and networks should allow users/data subjects to express privacy preferences and controls in a persistent and effective way.**7.3–Encourage Direct User/Data Subject Access:**Software systems should be designed to provide data subjects direct access to data held about them, wherever feasible, and an account of uses and disclosures. | **SHALL** describe user/data subject privacy options, including (access) controls, privacy preferences/settings, UI/UX supports, and user/data subject- centric privacy model**SHALL** describe notice, consent, and other privacy interactions at the EARLIEST possible point in a data transaction exchange with a user/data subject or her/his automated agent(s) or device(s)      |