Eco-system for disease specific clinical workflow and data integration

DELIVERABLE D2.5

Open Algorithm and Application Ecosystem Architecture for SYMPHONY

Project number:	ITEA 21026
Document version no.:	v 1.0
Edited by:	Yusuf Sayıta
Date:	May 2025

ITEA Roadmap challenge:
Smart Health

HISTORY

Document version #	Date	Remarks
V1.0		Final version

TABLE OF CONTENTS

1 INTRODUCTION 4

2 DEFINITION OF OPEN APIS AND INTEGRATION PROCESSES 5

2.1 APIs for Workflow Definition 5

2.2 APIs for Application or Algorithm Integration and Deployment 5

2.3 APIs for Algorithm Inference Workflows 6

2.4 APIs for Application Hosting and Launch Capabilities 7

2.5 APIs for the Feedback Loop 8

3 SECURITY AND PRIVACY SOLUTIONS FOR INTEGRATED COMPONENTS 10

3.1 Authentication 10

3.2 Authorization & Access Control 10

3.3 Data Protection 11
3.3.1 Encryption 11
3.3.2 Pseudonymization and Anonymization 11

3.4 Isolated Execution 12

3.5 Audit Logging 13

3.6 Regulatory Compliance 13

3.7 Security Standards 14

4 CONCLUSION 15

5 REFERENCES 16

1 Introduction

This document, Deliverable 2.5, serves as a blueprint for the secure and seamless integration of independently developed clinical applications and algorithms into the SYMPHONY ecosystem. It outlines the specific Open APIs, technical strategies, and operational processes required to expand the ecosystem's capabilities by integrating new components. This approach is fundamental to fostering innovation in digital health solutions.

The architecture is built upon foundational work previously documented. The concept of an open ecosystem with defined interfaces, where independently developed components can be integrated via prescribed rules, guidelines, and interfaces, is established. This emphasizes openness, allowing multiple vendors to provide components and avoiding vendor lock-in.

Key technical aspects necessary for integration are defined, drawing on comprehensive strategies and architectural designs. This includes the adoption of interoperability standards, such as HL7 FHIR for structured data exchange, DICOM for medical imaging data, and openEHR for long-term data preservation. Standardized terminologies like SNOMED CT and LOINC are utilized for semantic consistency. RESTful APIs and OpenAPI specifications (OAS) provide standardized methods for data exchange and clear API documentation.

The architecture incorporates a robust framework for data access, ingestion, and processing, detailing how integrated components interact with various data sources, modalities, and origins. This involves strategies for multi-vendor data integration, data transformation, validation, and quality control, supporting both real-time and batch processing through defined data pipelines. Data ingestion components include adapters and converters to standardize heterogeneous data formats.

Furthermore, stringent security and privacy requirements are integral to the architecture. This includes a multi-layer security architecture based on Authentication, Authorization & Access Control, and Audit Logging. Specific mechanisms like multi-factor authentication (MFA), Role-Based Access Control (RBAC), OAuth2, encryption (AES-256 for data at rest, TLS for data in transit), pseudonymization and anonymization techniques, and consent management are required. Adherence to standards like NEN7510:2020, ISO 27001:2023, GDPR, KVKK, HIPAA, and national privacy frameworks like Sweden's PDL and Spain's regulations is mandated where applicable. Security risk assessment and management strategies are also required for components. The architecture supports data access and portability rights in standardized formats. Robust vendor management processes are essential to ensure compliance of integrated components.

This comprehensive framework, integrating foundational architectural concepts, interoperability standards, data management strategies, and robust security and privacy measures, establishes the precise technical interfaces and processes. This ensures external applications and algorithms can securely, interoperability, and effectively operate within the SYMPHONY ecosystem, supporting crucial functions for diagnosis, treatment, and follow-up.

2 Definition of Open APIs and Integration Processes

The SYMPHONY project emphasizes the creation of an open ecosystem that fosters seamless integration of independently developed components. Central to this is the concept of a SYMPHONY API Ecosystem. Its core characteristics are interoperability, standardization, scalability, and security. This ecosystem includes Public APIs designed to allow external developers to extend SYMPHONY solutions with new components and capabilities. The architecture aims for cross-component interactions preferably via Public APIs, with private/internal APIs used only as a fallback, with potential to transition to public APIs. The SYMPHONY project will act as the governing body for these reference architecture APIs.

The definition of specific APIs and processes for integration points in D2.5 is based on the interfaces and architectural components identified in D2.1 is explained in the following sections.

2.1 APIs for Workflow Definition

These APIs relate to the interaction point between Workflow Management systems and Applications (Interface III in D2.1). While standards like CDS Hooks and FHIR Cast are noted for triggering based on defined events, D2.1 and D2.2&D2.3 explicitly state that no widely accepted and employed standard for defining workflows in systems like RIS or HIS could be found. D2.5 must therefore define the specific SYMPHONY approach and required interfaces for new components to interact with or potentially influence clinical workflows, addressing this identified gap.

The SYMPHONY Workflow API will support dynamic workflows driven by patient status, AI-recognized mentions, or questionnaire results, enabling the tailoring of care plans and follow-ups. For example, the API would allow for the initiation of new workflow sections (e.g., anticoagulation counselling or wearable ECG deployment) when specific clinical events are identified.

2.2 APIs for Application or Algorithm Integration and Deployment

This capability integrates independently developed "components" via defined interfaces. The SYMPHONY API Ecosystem includes Public APIs specifically for external developers to add new components. D2.5 defines the specific technical interfaces and processes for registering, making available (the marketplace concept), and deploying new components within the ecosystem. This includes defining interfaces for component lifecycle management.

Component onboarding will require metadata submission (e.g., intended use, regulatory status, runtime environment, input/output types), and a dynamic validation pipeline will assess compatibility and readiness.

An Application Registration API is proposed, providing secure endpoints for registering, updating, and removing applications. This includes a component manifest schema that must be submitted upon registration, facilitating automated integration testing.

The SYMPHONY Application Registration API will provide secure endpoints for registering, updating, and removing applications, facilitating seamless deployment and integration of third-party digital health components. The design of these marketplace APIs is informed by partner experiences in managing independently developed modules such as chatbots, dynamic care plans, and medication adherence tools.

2.3 APIs for Algorithm Inference Workflows

Integrated algorithms will interact with data sources and stores. This involves interfaces like Data Sources to Applications (Interface II in D2.1) and Data Stores to Applications (Interface IV in D2.1). Standards like CDS Hooks are mentioned for triggering algorithm execution or decision support based on events. The SYMPHONY project defines a data ingestion and processing architecture. This architecture is designed to handle and transform large amounts of healthcare data and make it ready for advanced analytics and visualization, which includes algorithm execution.

This architecture outlines a series of stages in data pipelines and workflows:

- **Ingestion Stage**: Raw data is captured from various sources such as Electronic Health Records (EHRs), medical imaging systems, laboratory results, IoT devices, and clinical notes. This data can be in diverse formats, including standard ones like DICOM and FHIR, as well as proprietary or unstructured formats like JSON, XML, text, PDF, or TIFF files.
- **Pre-processing Stage**: The ingested data undergoes cleaning, normalization, and transformation. This involves steps like removing duplicates, handling missing values, and converting data to standardized formats. Specific examples include using Natural Language Processing (NLP) techniques for clinical notes or image pre-processing for medical images.
- **Storage Stage**: After pre-processing, data is stored in data lakes for raw, unstructured data, or data warehouses for structured and processed data. The storage stage is typically kept closed to ensure data security and integrity, but APIs can be provided for controlled querying.
- **Processing Stage**: Data is further processed using either batch or real-time techniques. This stage can involve aggregating data, running machine learning algorithms, and performing complex transformations. An example cited is the analysis of ECG data to detect anomalies indicative of atrial fibrillation in real-time.
- **Analysis Stage**: Processed data is analyzed using machine learning models, statistical methods, and other analytical tools to extract meaningful insights, supporting clinical decision-making.
- **Visualization Stage**: Analyzed data is presented using visualization tools like dashboards and reports for end-users, such as healthcare professionals.

The processing stage, where algorithm execution occurs, is designed to allow external plugins, especially for machine learning and advanced analytics, facilitating the integration of new models and algorithms. D2.5 therefore defines the specific APIs, payload structures, and endpoints necessary to support the automated execution of algorithms, integrating them into these defined data processing pipelines.

To ensure uniformity across deployments, the Algorithm Execution API is proposed, which defines execution triggers, input schema referencing clinical data models (e.g., FHIR Bundles), execution context metadata (e.g., patient ID, timestamp, device), and output structure referencing standard vocabularies such as SNOMED CT or LOINC.

The proposed Algorithm Execution API will define execution triggers and input/output schemas to support automated execution of algorithms, such as those used to calculate risk scores or determine medication needs based on structured symptom data or transcribed conversation segments. Partner experiences integrating such inferencing pipelines into clinical timelines will inform the definition of SYMPHONY’s payload structures, execution triggers, and API endpoints for embedding algorithms into longitudinal patient pathways.

2.4 APIs for Application Hosting and Launch Capabilities

Components are defined as independently developed and released systems or services, which can integrate into a bigger solution. The integration of multiple components creates the SYMPHONY ecosystem, which is envisioned as a dynamic and user-friendly ecosystem/solution. The architecture supports a distributed, and extendible data infrastructure potentially based on cloud platforms and/or on-premises infrastructures in hospitals. This distributed infrastructure is intended to be scalable and support very large amounts of data for running analytics to help decision making.

The SYMPHONY project defines a reference architecture establishing a unified foundation for the development of components in the eco system and the interfaces between them. This architecture outlines how components categorized as Data Sources, Data Stores, AI Powered Components, and Application Components interact. The interfaces between these components are crucial for integration. The architecture includes Data Stores to Applications (Interface IV) and Applications to Data Stores (Interface V) interfaces, among others, which are fundamental for applications interacting with data within the ecosystem.

The data access, ingestion, and processing architecture, detailed in D2.4, further elaborates on the infrastructure considerations relevant to application hosting and launching. This architecture involves ingesting data from various sources such as Electronic Health Records (EHRs), medical imaging, laboratory results, IoT devices, and clinical notes. It emphasizes using cloud platforms (e.g., AWS, Azure) to provide scalable and flexible data storage and processing capabilities, facilitating access to data from various origins. RESTful APIs are implemented to allow different systems and applications to communicate and exchange data, including accessing patient records, lab results, and imaging data. OpenAPI specifications are used to define the structure and documentation of APIs, enhancing interoperability.

D2.5 therefore defines the specific technical interfaces that facilitate the hosting and launching of integrated applications within the SYMPHONY environment, leveraging the architectural discussions from D2.1 and these infrastructure and data management considerations.

The Launch Management API will allow SYMPHONY system operators to initialize, pause, resume, or terminate applications, with support for synchronous (immediate) and asynchronous (event-queued) launches.

Application launch configurations should include memory/CPU limits, environment variables, runtime dependencies, and fallback options for failure handling (e.g., retry policies, default timeout behaviour).

Security tokens for launch authorization will be issued through the identity provider components, using OAuth2 scopes to determine which services are allowed to launch which applications under which conditions. The SYMPHONY ecosystem utilizes OAuth2 for authentication and authorization, verifying user identities and issuing tokens that grant access to specific resources based on permissions. This approach reduces the need to share passwords and allows for granular control over access.

The Launch Management API will allow SYMPHONY system operators to initialize, pause, resume, or terminate applications securely within patient-specific contexts, supporting dynamic assembly of components like consultations, educational content, and clinical follow-ups. Partner infrastructure expertise will help define the technical protocols for launching applications on demand, maintaining state, and integrating identity, authorization, and consent flows. The security architecture of SYMPHONY is designed to ensure comprehensive protection through a multi-layered model including Authentication, Authorization & Access Control, and Audit Logging, supported by

encryption and pseudonymization. Role-Based Access Control (RBAC) mechanisms define user permissions based on role-specific requirements, ensuring users access only necessary data. Secure communication is ensured through TLS/SSL encryption to prevent unauthorized access. The project aims for compliance with regulations like GDPR.

2.5 APIs for the Feedback Loop

This section addresses the functionality for feeding results or decisions back into the SYMPHONY ecosystem. This process is critical for enabling the system to incorporate insights derived from data processing and analysis, closing the loop between algorithmic insight and clinical practice. This feedback can leverage architectural interfaces defined in the reference architecture (D2.1), such as Applications to Data Sources (Interface I), which is designed to share enhanced results back to the data source, or Applications to Data Stores (Interface V), used to send structured outputs to data stores for storage. The Data Store components within the architecture are responsible for storing data in a standardized fashion and facilitating its dissemination, supporting both clinical and research needs. Within this context, the concept of Clinical Decision Support (CDS) is relevant, particularly the potential use of standards like CDS Hooks for providing decision support or feedback. D2.5 therefore defines the specific APIs necessary for integrated components to feed clinical decisions, algorithm results, or other relevant feedback back into the decision support systems or relevant workflows.

The overall integration process for components that provide feedback involves defining their data flow paths, explicitly building on the data ingestion and processing architecture described in earlier project tasks. This architecture outlines stages such as ingestion, pre-processing, storage, processing, analysis, and visualization, noting which stages can be open for external plugins to enhance system capabilities. The design of the feedback loop APIs is informed by project partners' experience in translating AI outputs and application results into actionable clinical tasks and reintegrating them into the patient timeline using established standards like FHIR and OpenAPI.

The architecture adheres to widely adopted standards for API design and data exchange to ensure consistency and interoperability. Information regarding the selection rationale for standards such as RESTful APIs, OpenAPI specifications, FHIR, and DICOM is available in project documentation, including the reference architecture (D2.1) and documents detailing strategies for the adoption of interoperability standards (D2.2&D2.3) and the data access, ingestion, and processing architecture (D2.4). Further insight into the standards intended for use is provided in project standardization documents (D7.3/D7.5). D2.5 specifically defines how integrated components must utilize these standards at the defined integration points within the ecosystem.

Project documentation, such as the deliverable on Data Ingestion Tools (D5.1), identifies examples of External APIs and AI components intended for integration. These components, such as those providing sarcopenia metrics, MS lesion segmentation, or processing medical data via LLMs, interact with the ecosystem via defined interfaces. Information regarding the implementation of access control and security measures for data processing, as detailed in project documents like the deliverable on Data Privacy and Security Tools (D5.2), also underpins the secure operation of these APIs. This includes a multi-layered security architecture built on Authentication, Authorization & Access Control, and Audit Logging, supported by encryption and pseudonymization techniques. The ecosystem utilizes OAuth2 for authentication and authorization and implements Role-Based Access Control (RBAC)

to manage permissions, ensuring secure and interoperable data access. D2.5 defines the concrete interfaces required for these listed components to integrate and facilitate the feedback loop within the SYMPHONY ecosystem.

To implement a robust feedback loop, the Feedback Submission API is introduced. This API supports structured payloads that include provenance metadata (who/what produced the result), confidence levels, decision support tags, and optionally links to underlying source data. All feedback will be timestamped, versioned, and assigned a unique event ID for traceability, following HL7 FHIR Provenance.

Use cases for the Feedback Submission API include:

- Updating EHRs with diagnostic insights (e.g., risk scores).
- Suggesting clinical actions based on algorithm outcomes.
- Publishing results to audit logs or research repositories.

This feedback mechanism enables integrated components to feed clinical decisions, algorithm results (such as AF risk assessment algorithms or adherence patterns), or other relevant feedback back into the system, closing the loop between algorithmic insight and patient care. This feedback can trigger new sections in the workflow (e.g., lifestyle interventions) or be stored for audit and review.

3 Security and Privacy Solutions for Integrated Components

Ensuring that newly integrated applications and algorithms operate securely and comply with regulations is a critical requirement for the SYMPHONY ecosystem. The system is designed to adhere to a comprehensive security and privacy framework. This framework is detailed across D2.1 (Compliance Requirements), D2.2&D2.3 (Security and Privacy Requirements), D2.4 (Security and Privacy Considerations), and D5.2 (Data Privacy and Security Tools). D2.5 defines the technical solutions and requirements that integrated components must meet to conform to this framework.

Key security and privacy requirements and how integrated components must address them, referencing the detailed specifications in previous deliverables, include:

3.1 Authentication

All integrated components are required to adhere to SYMPHONY's ecosystem-wide framework for authentication and authorization, rather than relying on a single monolithic service. This framework is built upon a multi-layer security architecture that includes fundamental pillars like Authentication, Authorization & Access Control, and Audit Logging. This framework mandates Strong Authentication protocols, including Multi-Factor Authentication (MFA) where applicable, to ensure only authorized users or systems gain access to the ecosystem. Access control is managed through Role-Based Access Control (RBAC), restricting permissions based on defined roles and limiting access to only what is necessary. Mechanisms like OAuth2 and token-based authentication are utilized for validating user and machine identities and facilitating secure sessions, sometimes involving a trusted authorization server. These protocols are implemented across relevant parts of the architecture, such as the Privacy-Preserving Gateway, and are built upon APIs designed with hardened middleware for identity verification and prevention of unauthorized access [Section 3.1 original text]. The details regarding these mechanisms, encompassing specific security requirements, access control implementations, the use of encryption (such as AES-256 for data at rest and TLS/SSL for data in transit), pseudonymization techniques, and the specific application of standards like OAuth2 and RBAC, are defined and detailed across several project documents, including the Reference Architecture (D2.1), Strategies for Adoption of Interoperability Standards & Security and Privacy Compliance (D2.2&D2.3), the Data Access, Ingestion, and Processing Architecture (D2.4), and Data Privacy and Security Tools (D5.2). D2.5 therefore defines the concrete technical solutions and requirements that integrated components must meet to conform to this robust, ecosystem-wide security and privacy framework.

3.2 Authorization & Access Control

Authorization and access control within the SYMPHONY ecosystem are managed through a structured framework of mechanisms implemented across components, rather than being handled by a single monolithic service. Role-Based Access Control (RBAC) serves as a fundamental mechanism, ensuring secure data access and compliance with privacy regulations such as GDPR. RBAC defines permissions based on clearly defined roles, such as healthcare providers or administrative staff, ensuring that users are granted access rights aligned with their specific responsibilities and limiting access based on the principle of least privilege. This approach adheres to "Need to Know" principles by limiting data access strictly to what is necessary for specific tasks or roles. Dynamic access controls can be implemented to grant access to specific data elements based on the user's role and necessity.

Authorization also leverages mechanisms like OAuth2, which issues tokens granting users access to specific resources based on their permissions, further enhancing security by ensuring users only perform authorized actions and minimizing credential sharing. This access control framework is deeply integrated with comprehensive logging and auditing tools, which capture detailed records of all interactions with sensitive data, including access attempts, modifications, and sharing. Secure storage and analysis of these audit logs ensure traceability of all access and modifications and support compliance with regulations like GDPR. The design of secure access mechanisms, consistent access control policies, the use of RBAC, dynamic access controls, and the integration with audit trails are detailed across the project's documentation. The specific technical solutions and requirements that integrated components must meet to conform to this ecosystem-wide security and privacy framework are what D2.5 therefore defines.

3.3 Data Protection

3.3.1 Encryption

Within the SYMPHONY ecosystem, data protection is a critical focus, and comprehensive encryption measures are implemented to ensure the confidentiality and integrity of sensitive health information. The system mandates specific encryption standards for data at different stages of its lifecycle:

- Data at Rest: Data stored within the SYMPHONY ecosystem is protected using AES-256 encryption. This means that even if the storage media is accessed by unauthorized individuals, the data remains unreadable without the appropriate decryption keys. AES-256 is noted as a widely adopted and robust encryption standard suitable for sensitive healthcare information. This safeguard protects against potential threats such as unauthorized physical access to data centers or servers.
- Data in Transit: When data is transmitted between components or systems within the SYMPHONY ecosystem, it is secured using TLS/SSL protocols, specifically TLS 1.2.

These protocols ensure that data is encrypted during transmission across networks. This measure protects against interception or tampering by unauthorized parties during data exchange. The implementation includes the use of strong encryption algorithms and secure SSL/TLS certificates to authenticate and secure communications between systems.

These encryption standards, applied across all integrated modules, are fundamental to maintaining data confidentiality and integrity during communication between services and storage. They are integral components of SYMPHONY's multi-layer security architecture. The requirements and implementation details for applying these encryption standards are detailed across the project's documentation, including the Reference Architecture (D2.1), Strategies for Adoption of Interoperability Standards & Security and Privacy Compliance (D2.2&D2.3), the Data Access, Ingestion, and Processing Architecture (D2.4), and Data Privacy and Security Tools (D5.2)

3.3.2 Pseudonymization and Anonymization

In alignment with regulations such as the General Data Protection Regulation (GDPR) and Türkiye's Law on Protection of Personal Data (KVKK), SYMPHONY strictly adheres to the principle of data minimization in all personal data handling. This means only the minimum necessary data is collected and processed to achieve intended purposes, thereby reducing the risk of exposure and enhancing compliance.

To further safeguard sensitive health information, SYMPHONY employs techniques including dynamic pseudonymization and anonymization. These techniques are applied across relevant data models, such as Patient and Observation data, to protect user identities and facilitate secure data sharing.

For instance, sensitive identifiers like patient identifiers are replaced with unique but non-reversible tokens or pseudonyms. For pseudonymization, a mapping table may be maintained to associate pseudonyms with original identifiers, with access strictly controlled for authorized users and legitimate purposes. Techniques like generalizing or aggregating remaining quasi-identifiers are used to enhance anonymity where complete identification removal is not feasible or required.

Specific data types, including IMU sensor data and AI inference results, are anonymized or de-identified before sharing with third parties or processing outside the clinical environment to allow analysis without revealing identifiable information. This process involves replacing unique identifiers with synthetic values. While sources do not explicitly detail cryptographic hash algorithms for verifying anonymized data integrity in the provided excerpts, the system emphasizes data integrity through methods like encryption and audit trails. Traceability of data interactions is ensured through comprehensive audit logging.

The requirements for implementing these privacy-preserving techniques, including specific strategies for pseudonymization and anonymization, details on data minimization, and examples of their application to data like patient records, sensor data, and AI outputs, are found across the project's detailed specifications. These include discussions in the Reference Architecture (D2.1) regarding de-identification methods, in Strategies for Adoption of Interoperability Standards & Security and Privacy Compliance (D2.2&D2.3) covering privacy protection strategies and implementation examples, in the Data Access, Ingestion, and Processing Architecture (D2.4) outlining de-identification techniques and data processing requirements, and extensively in Data Privacy and Security Tools (D5.2) detailing the processes for pseudonymization, anonymization, data minimization, and their use within specific use cases.

3.4 Isolated Execution

To safeguard against the risks posed by the integration of various AI modules and external software components, the SYMPHONY ecosystem architecture is designed to treat these components as independently developed and released systems or services. The integration relies on prescribed rules, guidelines, and interfaces, defining how these separate entities interact within the unified solution. Ensuring the security of individual components is a fundamental requirement, alongside the security of the integrated system.

Components are expected to undergo security risk assessments and mitigate identified risks to conform with the architecture. Security audits are also systematically conducted at different levels. While specific real-time threat detection mechanisms exist within parts of the ecosystem, such as real-time intrusion detection and AI-powered anomaly detection in certain use cases, the overall multi-layer security architecture aims to provide comprehensive protection by monitoring and managing component interactions and potential threats.

Building upon the general security principles, risk mitigation strategies, and the multi-layer security architecture established within the project's documentation, specifically the Reference Architecture (D2.1) and Strategies for Adoption of Interoperability Standards & Security and Privacy Compliance (D2.2&D2.3), D2.5 therefore defines the concrete technical solutions and specific requirements necessary to ensure the

secure integration and operation of components. This framework is crucial for preventing malicious code or errors in one component from adversely affecting others within the ecosystem.

3.5 Audit Logging

Within the SYMPHONY ecosystem, a robust Audit Logging framework is a mandated requirement for all integrated components. This framework is essential for establishing and maintaining transparency, accountability, and security. The system design requires that all interactions with sensitive data be captured and recorded in detailed logs.

These logs are required to track various activities, including access attempts, modifications, data transfers, and sharing. They are designed to capture relevant information such as user activities, timestamps, and potentially IP addresses.

The primary purposes of this comprehensive logging are traceability, security monitoring, and forensic analysis. By maintaining a detailed record of all interactions, the system enables administrators to monitor usage, detect suspicious or unauthorized behavior in real-time, and facilitate investigation in the event of a security incident. Audit logging is also a critical mechanism supporting compliance with privacy regulations like GDPR.

This comprehensive logging mechanism serves as a fundamental pillar of the multi-layer security architecture, ensuring traceability of all data-related actions. The requirements for implementing this audit logging framework, including details on the information to be captured and its role in overall system security, are foundational elements defined within the project's security and compliance documentation.

3.6 Regulatory Compliance

Adherence to stringent legal and regulatory frameworks is a foundational requirement for the SYMPHONY ecosystem. The architecture is designed to ensure robust compliance with major data protection laws and standards applicable to health data management and medical software. This includes alignment with:

- The General Data Protection Regulation (GDPR), which governs privacy and data protection aspects, focusing on user consent, data minimization, and security.
- Türkiye’s Law on Protection of Personal Data (KVKK), requiring adherence to requirements for personal data acquisition, management, transfer, and processing within Türkiye, including obtaining direct consent from patients and anonymization when processing reasons are eliminated.
- HIPAA, alongside GDPR, is highlighted as a key regulation for safeguarding patient data from unauthorized access and misuse.
- Relevant national privacy frameworks, such as Sweden's Patient Data Law (PDL), which includes specific requirements for data storage, traceability of access (logging), access restrictions, and data deletion processes. Compliance with local regulations in each country where medical applications operate is mandated
- The Medical Device Regulation (MDR) for components classified as medical devices, requiring conformity assessments, clinical evaluations, and adherence to post-market surveillance requirements to ensure safety and effectiveness. Each classified medical device component must comply with MDR.

The SYMPHONY architecture requires that each component integrated into the ecosystem must be validated for regulatory compliance according to the applicable regulations in the country where it is used.

Ensuring compliance also involves implementing security risk assessment and management strategies for components. Components are expected to undergo security risk assessments and mitigate identified risks. Security audits are systematically conducted at different levels, including software code, application, and APIs.

For new integrations, adherence to prescribed rules, guidelines, and interfaces is required. Robust vendor management processes are essential to ensure that all integrated components comply with relevant privacy regulations and security requirements.

To identify and mitigate risks associated with data processing activities, the architecture requires the implementation of Data Protection Impact Assessments (DPIAs).

Furthermore, aligning with regulatory mandates and patient rights, the system is required to support data access and portability rights. This includes the capability for users to access their personal data upon request and download or transfer their data in structured, machine-readable formats.

These stringent requirements for regulatory compliance, including specific regulations, risk management, vendor validation, DPIAs, and data portability, are fundamental to safeguarding sensitive health data, maintaining patient trust, and are comprehensively defined within the project's documentation on security and compliance.

3.7 Security Standards

Adherence to relevant security standards or the principles they embody is a fundamental requirement for all integrated components within the SYMPHONY ecosystem. This approach is crucial for ensuring the integrity, confidentiality, and availability of sensitive health data. While specific component deployments, particularly within hospital infrastructure environments, are expected to comply with standards such as NEN7510:2020 and ISO 27001:2023, and internal software development within partner organizations adheres to ISO 27001, the core principles of these and other recognized security frameworks are integral to the overall system design.

Each component is required to undergo security risk assessments and mitigate identified risks to conform with the architecture. Security audits are systematically conducted at various levels, including software code, application, and APIs. This focus on standards and principles ensures that components contribute effectively to the system's multi-layer security architecture, which is built upon pillars such as authentication, authorization & access control, and audit logging, supported by mechanisms like encryption and pseudonymization. The architecture is designed to comply with relevant regulations and standards, including GDPR.

Building upon the general security principles, risk mitigation strategies, and the security architecture established within the project's documentation, specifically the Reference Architecture (D2.1) and Strategies for Adoption of Interoperability Standards & Security and Privacy Compliance (D2.2&D2.3), D2.5 therefore defines the specific security standards and requirements that integrated components must meet to ensure the overall integrity and security of the SYMPHONY ecosystem.

4 Conclusion

The SYMPHONY ecosystem's architecture is fundamentally designed to enable the secure and seamless technical integration of diverse components, including new applications and algorithms. This open approach is pivotal for fostering innovation and expanding the capabilities of digital health solutions within healthcare workflows.

The architecture mandates specific technical requirements that integrated components must satisfy to participate effectively in the ecosystem. Key among these is the adherence to standardized interfaces and Open APIs, which ensure consistent and compatible interactions. This includes leveraging established standards for data exchange and storage, such as HL7 FHIR for structured clinical data and DICOM for medical imaging. The architecture also supports openEHR for flexible and comprehensive health data representation and long-term integrity.

Central to this architecture are robust security and privacy requirements. Integrated components must implement measures such as data encryption (at rest and in transit using standards like AES-256 and TLS 1.2), strict access controls (including Role-Based Access Control - RBAC), and audit logging to track interactions with sensitive data. Pseudonymization and anonymization techniques are also required to protect sensitive patient data, particularly for research or analysis purposes.

Furthermore, the architecture is built upon a foundation of regulatory compliance. It is designed to align with major data protection laws and standards like GDPR, KVKK, HIPAA, and national frameworks such as Sweden's PDL. Components classified as medical devices must also adhere to regulations like the Medical Device Regulation (MDR). Security risk assessment and management are required for components. The system is also required to support data access and portability rights, enabling users to securely download and transfer their medical data in standardized formats. Robust vendor management processes are essential to ensure integrated components comply with these privacy regulations and security requirements.

This comprehensive architectural framework, encompassing defined Open APIs, integration processes, robust security, privacy, and compliance requirements, is fundamental to the SYMPHONY project's goal of creating an open healthcare IT ecosystem. By enabling the integration of diverse components, the architecture supports the core objectives of providing care professionals with real-time, comprehensive insights into a patient's status, facilitating efficient decision-making for diagnosis, treatment selection, and follow-up, and ultimately improving patient outcomes.

5 References

[1] SYMPHONY Consortium, "SYMPHONY Deliverable 2.1 - Reference architecture for open eco-system," 05 01 2024. [Online]. Available: https://itea4.org/project/workpackage/deliverable/document/download/223/SYMPHONY%20D2.1%20Reference%20architecture%20for%20open%20eco-system.pdf. [Accessed 01 05 2025].

[2] SYMPHONY Consortium, "SYMPHONY Deliverable 2.2 & 2.3 Strategies for Adoption of Interoperability Standards & Security and Privacy Compliance," 27 03 2024. [Online]. Available: https://itea4.org/project/workpackage/deliverable/document/download/256/SYMPHONY%20D2.2&D2.3%20Strategies%20for%20adoption%20of%20interoperability%20standards.pdf. [Accessed 01 05 2025].

[3] SYMPHONY Consortium, "SYMPHONY Deliverable 7.3 & 7.5 Standardisation & Dissemination Plan," 30 10 2023. [Online]. Available: https://itea4.org/project/workpackage/deliverable/document/download/184/SYMPHONY%20D7.3,%20D7.5%20Standardisation%20&%20Dissemination%20Plan%20v1.0.pdf. [Accessed 01 05 2025].