Work Package	**WP5.1 - GenDAI Interactive Reporting - Visual Data Analysis**
Task(s)	**T5.1 Visual structures** **T5.2 Addressing user needs**
Deliverable	**D5.1 GenDAI Interactive Reporting Visualisations:** The identified visual structures will be reported according to the user needs, the operating scenarios, and the analytical tasks identified in WP1, and the different identified users.
Author(s)	Paolo Buono, Thomas Krause, Rosa Lanzilotti, Pasquale Ardimento, Philippe Tamla, Flavia Monti, Francesca De Luzi, Massimo Mecella
Version	1.0
Dissemination Level	PU

Table of Contents

Executive summary.. 3
1. Introduction.. 4
1.1 Overview.. 4
1.2 Purpose of this Document and its Organization.. 4
2. Diagnosis, analysis, IBD.. 5
2.2 Use Cases.. 6
2.3 Data.. 8
3. State of the art of reports, visualizations, and visual analytics.. 9
3.1 Evolution of Microbiome Diagnostic Reports.. 9
3.2 Visual Representation of Microbiome Data.. 10
3.3 Visual Analytics for Clinical Interpretation.. 11
3.4 Technologies for dynamic reporting.. 11
3.5 Interactive PDFs.. 16
3.6 Libraries and Technologies.. 18
3.7 Java and the Programmatic Management.. 21
3.8 Standards, Compliance, and Validation.. 24
3.9 Advanced Techniques for Interactivity and Personalisation.. 27
3.10 Production Workflows.. 30
3.11 Testing and Experimental Validation Tools.. 32
3.12 Conclusions.. 35
4. Interactive reporting and visualizations.. 36
4.1 Existing works.. 37
4.2 A report prototype.. 44
4.3 Assessment of the first version.. 47
Conclusions and Next Steps.. 49
References.. 49

Executive summary

Overall Assessment

WP5 aims to deliver innovative visual user interfaces and interactive clinical reporting (GenDAI Interactive Reporting). The main results presented in this first WP5 deliverable are related to the development of dynamic reports that the identified end users (clinical pathologists, doctors and patients) will use. In this WP we have addressed three use cases that are relevant for the analysis of the samples, exploiting visualizations and interactivity in the document.

Through the visualisation of data produced by sequencing processes, GenDAI addresses several needs: 1) provide patients with interactive visualisations to improve the understandability of analysis result; 2) provide domain experts with advanced visual tools to perform comparison among different analysis and understand the available data; 3) visualise the partial computation of the data in the cases where such computation goes beyond the perceived loss of interaction.

This deliverable introduces a novel way to present the reports to the end users that goes beyond the current practices to create static reports that contain the analysis results. Being at the end of the analytical pipeline, it represents a bridge between the user and the system; it is the visible part through which the system reveals itself to the user, and it is the means with which the user communicates with the system.

This approach is innovative for the analyzed domains (biology and microbiome) but has the potential to improve all fields where static reports composed of multiple pages containing items that require explanation are needed.

Outputs and Project Progress

This deliverable presents early results, which will be extended and finalized by month 18, demonstrating that the current direction is promising. The proposed dynamic finding report exploits interaction and visualizations to provide in a very short time the results of the sample analysis, organized by priority. The proposed approach contributes reaching the KPI defined in the RIO stated in table 1 of the DOA, more specifically RIO 5 (Deliver innovative visual user interfaces and interactive clinical reporting-GenDAI Interactive Reporting).

Contributions to Impacts

The contribution presented in this deliverable is inline with the DOA from different perspectives. The user will benefit from an improved organization of the reports delivered after the analysis of the patient's stool; The clinical pathologist and the data analyst will benefit from the same structure of the report because they will have the possibility to quickly select the data of interest. The capability of the report to include more than one patient and to compare different patients will empower users (clinical pathologist and data analyst) to better find relationships among patients' results and valuable patterns in search for medical solutions to patient´s aliments.

These results are desirable from a scientific as well as commercial standpoints, because substantially better lab reports will enhance next steps in treatment while offering additional insights, become more engaging while reducing errors and overall processing time. The system component is built on the IVIS4BigData reference framework, which has been demonstrated as a valid reference model capable of managing big data and supporting visualizations.

1. Introduction

1.1 Overview

The GenDAI project (Genomic applications for laboratory Diagnostics supported by Artificial Intelligence) aims to develop a novel medical diagnostics platform that leverages metagenomic data and Artificial Intelligence (AI) to improve the diagnosis and monitoring of inflammatory bowel disease (IBD). The platform will provide clinicians with a comprehensive, personalized view of a patient's microbiome, enabling more targeted and effective treatment strategies. The medical diagnostics platform leverages AI and metagenomic data for faster, more accurate clinical analysis. The platform automates data workflows to enable personalized medicine, initially focusing on tangible benefits for conditions like Irritable Bowel Syndrome (IBS).

This deliverable focuses on three types of user, identified in early stages of the project: end-user, clinical pathologist and data analyst. The end-user is the patient that provides the stool for the analysis, but is also the physician that reads the results and provides indications and the therapy to the user. The clinical pathologist and the data analyst are people that analyze data to provide the patient a therapy and to support the clinical pathologist in the analysis of the problem inter-patients.

This deliverable also focuses on two of the three main aspects being covered in this project, that are both related to the creation of dynamic reports. According to the user, the dynamic report shows the analysis results of a single patient, or shows the results of multiple patients with the goal of the comparison among them. In the remaining part of the project, the focus will be devoted, initially to assess the validity of the proposal, according to the KPIs identified in the RIO, then the focus will move towards the clinical pathologist and the data analyst to support the computation of the analysis.

The current practice in the analysis laboratories is to provide the user with static reports. In this project we provide dynamic reports that take into account the needs of the end user, by providing immediately the information that requires attention, and the less relevant information in the next pages of the report. To address the information request about the user, the traditional reports contain a separate area that explain the meaning of the analyzed parameters. In this project, the reports are built in different formats (web, interactive pdf, static pdf) to allow the user to jump in the area where the explanation is reported to get immediately to the answer to the users' questions.

1.2 Purpose of this Document and its Organization

This document describes the contribution offered by GenDAI related to the data visualization, report generation, and presentation to the end user.

Section 2 introduces the domain of IBD, the diagnosis procedure, and the involved users. Section 3 reports a state of the art related to existing diagnostic software and reports, with an emphasis on the reporting and the analysis of the data and presents the technologies that can be used to generate reports that can be both static and dynamic. Section 4 of the document illustrates existing works and the proposed dynamic report.

2. Diagnosis, analysis, IBD

2.1 Introduction

Genomics-based diagnostic data (GBDD) are becoming increasingly important for laboratory diagnostics. Due to the large quantity of data and their heterogeneity, GBDD poses a big data challenge. Current analysis tools for GBDD are primarily designed for research and do not meet the requirements of laboratory diagnostics for automation, reliability, transparency, reproducibility, robustness, and accessibility. This makes it difficult for laboratories to use these tools in tests that need to be validated according to regulatory frameworks and to execute tests efficiently, in terms of time and cost. To better address these requirements, this project proposes an architecture inspired by Krause et al. as the basis for supporting the analysis [1]. This deliverable primarily focuses on the User Interface of the end-user, and specifically the UI related to the reports generated after the analysis. The UI is only the last step of a process of acquisition, transformation, and validation of the data that are complex, big, and needs to be quickly processed, which fits in the field of Big Data. In this context, Bornschlegl et al. introduced the IVIS4BigData Reference Model to standardize artifacts and process steps for combining the research domains of Big Data and AI and reduce the complexity for future applications [2]. The addition of visualization further strengthens the field because whatever system people use, they very likely will interact with it through visual interfaces, and the best medium that should be used to convey a large quantity of information in a limited quantity of time is visually, which exploits in parallel the analytical possibilities. Indeed, textual data, as well as audio data, require a sequential information transfer, and touch requires a 2D/3D scan of the surface that contains information, which is even worse in terms of efficiency. The visual means convey a lot of information at the same time, but must be used correctly to optimize the process.

Big Data, AI and visualizations, combined together, can provide superpowers to the user who might be provided with tools and could perform use cases that could not have been implemented otherwise[3].

In genomics, there is a strong need for visualization, due to the high dimensionality, complexity, and heterogeneity of data, and numerous researchers have attempted to utilize visualization in metagenomic analyses [4]. GutMeta comprises over 90 projects and provides users with several visualization techniques to reveal relationships among biological data. Specifically, it shows a correlation matrix and a radial tree as central techniques [5]. Another technique employed is the horizon chart technique, which consists of a compact stacked area chart representation that helps to find patterns over time [6]. In general, a single visualization is insufficient to support the analysis process; therefore, multiple coordinated visualizations are necessary in this context [7,8]. This is also reflected by existing surveys on biovisualization that mostly focus on specific aspects of the data or specific biological analysis tasks. Kerren and Schreiber [9] presented a survey focused on network visualization, with a specific focus on cellular networks. They review existing biological approaches and highlight how analysis and visualization tools for biological networks are increasingly available. One example is the KEGG pathway maps that detail molecular interactions and reactions [9,10]. Another survey focused on visualizing medical data using glyphs [11]. They provide six guidelines for the usage of the glyph technique. Suschnigg et al. [12] propose a multivariate synchronized visualization that also uses glyphs to visualize anomalies in cyclic time series data. Even if the domain they explore is in the automotive sector, their tool can be adapted to the biological domain.

When high-dimensional data must be analyzed, glyphs can be combined with dimensionality reduction techniques such as principal component analysis (PCA) or t-distributed stochastic neighbor embedding (t-SNE) [13]. These visualizations are important for research, but are not adapted for visualizing GBDD of individual patients. Traditionally, results from medical laboratories are visualized only in a very simplistic way, if at all. For example, many laboratory parameters have a reference range, and the visualization is limited to representing this reference range along with the measured value. However, such simplistic visualizations quickly reach their limits with GBDD, since many results cannot be represented in 1D. Moreover, software for creating laboratory reports is often a fixed part of laboratory information management systems (LIMS) and is thus only adaptable and expandable to a limited extent.

The size and complexity of metagenomic data require adequate tools to support analysts and end-users. One of the most effective information seeking strategies is the Visual Analytics (VA) Mantra: “Analyse First - Show the Important - Zoom, Filter and Analyse Further - Details on Demand” [14]. It well applies in Big Data context and focuses on the interaction between human and machine. When data is big or complex, performance is important. Zgraggen considers three types of performance related to visualization: instantaneous, blocking and progressive [15]. Instantaneous occurs when the system reacts immediately. The blocking type occurs when the machine stops the interaction to compute the answer, such as a progress bar. The progressive type, instead of letting the user wait the entire data transfer, or computation, the system anticipates the estimated result, in order to reduce the time to take decisions.

The way the machine gets the results might be unclear to the user, generating a lack of trust. Bioscience strongly needs visualizations because of the dimension, complexity and heterogeneity of the data biologists are working with. Attempts to use visualization in bicrobiome analysis have been done by many researchers. Molecular genetic microbiome data have similarities with Electronic Health Records (EHR). Visualizations in EHR can be ported into the biological domain. Rind et al. analyze InfoVis systems and study how to make recommendations of visualization of EHR systems according to various dimensions [16]. Ten years later, a recent survey revealed that this domain is dominated by glyph and standard 2D (bar chart, line chart and pie chart) visualization techniques [17]. Many visualizations use glyphs that can be combined with dimensionality reduction techniques when high-dimensional data must be analyzed. Glyphs adopt metaphor and abstraction to help people find interesting patterns.

When interacting with AI, users must understand its results and reasoning. Lengthy reports often necessitate users to browse back and forth, causing cognitive strain. Explainable AI adoption depends on the trust the user has on a system [18], which increases with the knowledge and the understanding of background processes often hiding in a system. By revealing data provenance [19] the user trusts more the validity of the results. Despite it is acquiring maturity, the VA literature still does not adequately consider recent and popular baselines, such as LIME [20] Grad-CAM [21] and SHAP [22].

GenDAI will use interactive reports to streamline interpretation, diminish mental effort, and expand the potential for discovering new insights. Aligning with contemporary literature such as GutMeta, GenDAI will introduce multiple coordinated visualisations [23] and will adapt visualisations targeted for Electronic Health Records (EHR) to interactive microbiome reports. To help users understand results and the underlying processes, especially in the context of AI methods, GenDAI will provide explanations through visualisations that integrate explanations in Natural Language using AI. This will help users build a mental model, thus improving effectiveness and efficiency.]

2.2 Use Cases

With reference to the use context reported in Section 3.1 of deliverable D1.1, in this document, the UCs considered are those related to UC3 Analyze Diagnostic Test Data, UC4 Create Reports, and UC6 Use Report. Some connections with UC2 are present, related to the analytical pipeline to analyze the samples, but this is the focus of the next deliverable (D5.2).

2.2.1 UC3 Diagnostic Test Data Analysis

The main actor in this use case is the data analyst, who receives the raw diagnostic data and performs manipulation activities (data preprocessing) to make the data ready to be computed by the machine. For repetitive tasks, it is automated and goes through established data transformation pipelines. The goal of the use case is to interpret patterns, anomalies, perform checks, and, if needed, consult domain experts. The findings are used to produce the diagnostic report.

Accuracy is addressed through several steps:

- Using controls (positive/negative) during the sequencing run.
- Monitoring machine-reported quality metrics.
- Applying cutoffs for minimum sequencing depth per sample; samples below the cutoff are discarded and re-run.
- Technical validation by lab staff checking machine performance and data quality.
- Medical validation by clinical pathologists reviewing the results for plausibility before release.

The validation process is specific in the laboratory and is not strictly related to the User Interface of GenDAI. Indeed, from the interviews emerged that the most significant barrier to widespread diagnostic use is the lack of a widely accepted consensus on the diagnostic process itself—particularly in sample preparation (e.g., DNA extraction), as well as the subsequent bioinformatics analysis and result interpretation.

The first part of the work in GenDAI is addressing the data visualization, the second part will consider the UI related to the laboratory experts, because from the interaction with the user and from the interview it emerged that the individual diagnosis processes are time-intensive and often include hands-on or overnight steps. As a result, long turnaround times for samples remain an issue, particularly in metagenomic whole-genome sequencing, and to a slightly lesser extent in 16S sequencing or PCR-based diagnostics. Given the challenges in wet lab work, the users aim for the most user-friendly software solutions. Indeed, the same staff handling the wet lab processes may also be using the software created in GenDAI.

2.1.2 UC4 Report Creation

This use case involves data analysts and clinical pathologists. Data analysts generate the preliminary report summarizing the key results, the interpretations, and the recommendations. The clinical pathologist reviews the report, checks for consistencies and errors, and finalizes the report for its delivery. Here, the creation of the report occurs; after this step, the report is locked and cannot be further modified.

2.2.3 UC6 Report Use

This use case is devoted to the end user, which includes physicians and patients. In general, the target user of the report is a practitioner seeking diagnostic information for patient care. It means that the report is intended for an informed person regarding the parameters shown. However, the report is also read by the patient or the caregivers, so it is important to provide information also for people who would search such information outside the report.

The case study starts upon receiving the report in both dynamic and static form. Here, the user must be taken into account regarding the content of the report. Since the report can be lengthy, the most important information should be provided in the first pages. This is a requirement that emerged during the interviews with the domain experts. The information must be provided in both graphical and textual ways. The graphical form helps see the information rapidly, while the textual form explains the results and avoids misinterpretations of the results. Both are necessary to have an efficient and effective means of providing information.

2.3 Data

2.3.1 Data collected from the patients

The analysis laboratory could require information about the disease state via the sample entry system. For gut microbiome testing, mandatory information includes age and sex (there are no gender-specific reference values yet) and sample collection date, optionally, diagnosed diseases, medication history (antibiotics, PPIs, antidepressants, etc.), lifestyle, eating habits, and dietary habits. Additionally, often other parameters from the stool samples, like inflammation markers, pH, and consistency, can be determined to provide context. Quality metrics from the sequencing run (e.g., reads per sample, quality scores) are also saved.

Data format standardization is important. While the core sequencing data within standard formats like FASTQ is consistent, variations exist, such as in header information or tags added by different sequencing platforms (e.g., Illumina vs. others). Within the bioinformatics community, common formats are generally used for specific techniques, reducing major inconsistencies for the initial raw data (FASTQ). Inconsistencies might arise more in the bioinformatics methods applied (e.g., OTU vs. ASV approaches).

Raw data produced for the 16S V3-V4 sequencing consists of two FASTQ files (forward and reverse reads) per sample. Each file is approximately 190 megabytes uncompressed, totaling around 360-400 megabytes per sample for the raw sequencing data.

For reference databases, there are public ones, such as SILVA and Greengenes, but laboratories may prefer to use their internal database.

SILVA provides quality-checked and regularly updated datasets of aligned small (16S/18S, SSU) and large subunit (23S/28S, LSU) ribosomal RNA (rRNA) sequences for all three domains of life (Bacteria, Archaea, and Eukarya).

Greengenes is a dedicated full-length 16S rRNA gene database that provides users with a curated taxonomy based on de novo tree inference.

Raw data are useful for the computation, but do not appear in the final report. FASTQ format files contain a huge quantity of information, but they also contain noise due to the sampling applied. So the pipeline requires the initial check of the sequence quality, data cleaning (denoise), sequence grouping to have the most significant taxonomic units, and finally the production of the abundance table that reveals the microorganisms quantities for each sample.

2.3.2 Data included in the report

The current version of the report (described in Section 4) contains basic patient's data (name, gender, age), when it is built for patient delivery, otherwise data that can be associated to the patient are removed; it also contains stool data, such as stool properties, biodiversity, enterotype, dysbiosis index, ratios (e.g. firmicutes/bacteroidetes, actinobacteria/proteobacteria, prevotella/bacteroides). It also contains phyla, bacterial phyla (most important genera and species), results, and therapy options, and metabolome.

The listed data are those that must be inserted in the report, both for the patient and for the physician. All such parameters require a description of the meaning for each one, and the relationship with other relevant parameters analyzed and appearing in the report. Indeed, the report should show the data, both in numbers and with visual clues that increase the understanding speed.

The last part of the report should also contain a verbose description of the results that provide useful information both for understanding and recommending therapies, also in the form of prebiotics and probiotics recommendations.

2.3.3 Data stored in GenDAI

In GenDAI, the datastore is intended to consolidate the different categories of information produced throughout the diagnostic workflow, integrating both technical data from sequencing activities and the analytical outputs required for generating clinically meaningful findings. At its current stage of design, the datastore is expected to include the raw files originating from sequencing runs, together with associated quality-control metrics and run-level metadata, ensuring full traceability of the laboratory process. It will also store the processed artefacts generated by the bioinformatics pipeline, such as taxonomic assignments, abundance tables, and other intermediate outputs that may be required for re-analysis or validation. Alongside these, the system will maintain the derived indicators used within the findings report, biodiversity measures, dysbiosis indices, relevant microbial ratios, and other interpretation-oriented values, together with the metadata necessary to contextualize them, including sample information (e.g., identifiers, age, gender, and sample collection date) and, where applicable, patient-level attributes. In support of reporting and visualization, the datastore will also host the structures needed for comparative analysis, such as reference cohort summaries or normative ranges. Furthermore, compliance, versioning, and audit information will be preserved to ensure reproducibility, long-term accessibility, and alignment with regulatory requirements. While this structure provides a solid foundation for the expected data flow, the precise organization and formats may be refined as the platform evolves and as additional requirements emerge during implementation and integration activities.

3. State of the art of reports, visualizations, and visual analytics

3.1 Evolution of Microbiome Diagnostic Reports

Diagnostic reports for gut microbiome assessment have evolved in parallel with the scientific understanding of **Irritable Bowel Syndrome (IBS)** and **Inflammatory Bowel Disease (IBD)**. Early foundational work in *Digestive and Liver Disease* and *Gastroenterology* documented the clinical burden of IBS, a functional gastrointestinal disorder characterized by abdominal pain, bloating, and altered bowel habits, affecting up to 10–15% of adults worldwide [24], [25]. These studies revealed the disorder's multifactorial nature and highlighted the limitations of symptom-based diagnosis in the absence of reliable biomarkers. As research progressed, evidence from high-impact journals such as *Microbiome*, *mSystems*, and *Gut Microbes* demonstrated that IBS is frequently associated with measurable dysbiosis, including reduced microbial diversity, shifts in dominant bacterial phyla, and reduced levels of key beneficial taxa such as *Faecalibacterium prausnitzii* and *Roseburia* spp. [26-29].

Despite these scientific advances, microbiome reports produced by clinical laboratories or commercial platforms remain static. They typically present macroscopic stool properties, summary indices (e.g., Shannon diversity, Simpson index), phylum ratios, and taxonomic abundance profiles, but often without explanatory context. As emphasized in methodological guidelines such as the STORMS checklist [30], microbiome data require structured, transparent reporting to avoid misinterpretation, yet most existing documents provide little narrative support to help readers understand the clinical meaning of reported values. Moreover, clinicians and patients often struggle with the technical vocabulary, statistical indices, and lack of clear guidance regarding what constitutes a clinically relevant deviation.

Commercial services attempted to address these gaps. uBiome offered consumer-oriented microbiome reports with simplified graphical elements and color-based alerts [31], but subsequent investigations revealed methodological flaws, inconsistent “healthy ranges,” poor transparency, and rigid presentation structures [32], [33]. Viome proposed metatranscriptomics-based reports with aggregate “health scores” [34], but the simplified scoring model was criticized by both clinicians and researchers for lacking clarity on how the metrics were derived. Even high-quality scientific service providers, such as those referenced in [35], generated reports rich in data but not optimized for accessibility by non-specialist readers.

Other medical fields demonstrate more mature reporting practices. In respiratory diagnostics, ResMed AirView integrates descriptive explanations with visualizations of apnea–hypopnea patterns and contextual indicators that guide interpretation [36]. Although unrelated to microbiome research, this system illustrates how user-centered clinical reports can improve interpretability by combining explanatory text, graphical summaries, and stratified clinical indicators.

Overall, the current state of microbiome reports reveals a gap between sophisticated scientific data and limited, static reporting formats. Without interactive elements, contextual explanations, or reference cohort comparisons, existing reports constrain broad clinical adoption and remain difficult for patients and non-specialist physicians to interpret.

3.2 Visual Representation of Microbiome Data

Visualizations are a central component of microbiome reports, serving as a bridge between raw data and clinical or patient understanding. Standard graphical formats, such as taxonomic bar charts, biodiversity gauges, abundance heatmaps, or phylum ratio diagrams, are widely used to summarize microbial community patterns. These visualizations can expose highly relevant IBS-related features such as reduced richness, altered Firmicutes/Bacteroidetes ratios, or depletion of SCFA-producing microbes, trends repeatedly documented in leading journals [26-28], [37], [38]. However, as noted by methodological reviews [30], [39], most microbiome visualizations lack interpretive framing. Graphs are frequently displayed without accompanying explanations or clinically meaningful narratives. As a result, users unfamiliar with microbiome ecology struggle to understand whether a displayed value is typical, concerning, or clinically irrelevant. Even experienced clinicians often face difficulties interpreting taxonomic visualizations, particularly when no thresholds, ranges, or comparison groups are provided [40]. The analysis of existing microbiome reports conducted in the GenDAI project confirms these gaps: data visualizations are often static, non-interactive, and not ordered according to clinical priority. Interview feedback showed that users prefer visualizations that foreground abnormal values first, while making “in-range” data secondary. Current microbiome reports do not support this preference, leading to confusion and slower identification of clinically relevant anomalies. In contrast, respiratory diagnostic tools such as ResMed AirView [33] offer a far more advanced model of clinical visualization, where textual explanations accompany charts, temporal trends are clearly displayed, and clinically meaningful categories (e.g., supine vs. non-supine sleep) structure interpretation. This type of visualization design, based on contextual, layered, and narratively supported, has not yet been adopted in microbiomics. The consequence is clear: while microbiome visualizations are abundant, they do not yet meet the usability, clarity, or interpretability standards established in other areas of digital medicine.

3.3 Visual Analytics for Clinical Interpretation

Visual analytics integrates interactive visualization with computational analysis, enabling users to explore complex datasets dynamically rather than passively receiving predetermined outputs. In the context of microbiome diagnostics, visual analytics is particularly relevant due to the high dimensionality and variability of microbial communities, the need to correlate taxa with symptoms, and the demand for comparison across personalized and population-level profiles. Existing microbiome reports seldom implement visual analytics principles. Static formats do not support dynamic filtering, anomaly-driven exploration, or user-adaptable narratives. Reports also rarely provide mechanisms for comparing results with demographically or clinically matched cohorts. This limitation is notable because cohort contextualization is essential for distinguishing normal variability from clinically meaningful deviations, as demonstrated in studies linking microbial signatures to IBS severity [41] and in broader analyses of microbiome-associated diseases [36]. From a technological perspective, advances in document engineering provide new opportunities for integrating visual analytics within widely used formats such as Portable Document Format (PDF). The evolution of PDF from a static container to a dynamic, programmable, and semantically enriched environment has been formalized through ISO 32000-1 and ISO 32000-2. These updates enable the inclusion of JavaScript-based interactivity, RichMedia elements, structured metadata, and accessibility-compliant semantic layers defined in standards such as PDF/UA. Libraries such as Apache PDFBox, iText, and OpenPDF provide programmatic access to these features, while validation frameworks like veraPDF and PAC 2021 ensure conformance with archival and accessibility requirements. Despite these technological capabilities, most microbiome reporting pipelines still rely on static documents that do not exploit the interactive or analytic potential of modern formats. This stands in contrast to trends in digital healthcare documented in high-impact journals such as Frontiers in Public Health, which emphasize the transition toward intelligent, adaptive, and context-aware interfaces [42]. The lack of interactive visual analytics in microbiome reporting, therefore, represents an unresolved challenge that limits the translation of complex biological data into actionable clinical insight.

3.4 Technologies for dynamic reporting

This section analyzes the possibilities offered by the PDF for the creation of interactive documents. Starting from the definition of the object model that constitutes the internal structure of PDF, the section explores the transition from static to dynamic documents, capable of integrating interactive elements, multimedia content, and scripting logic.

The evolution of the standard, formalised through ISO 32000-1:2008 and ISO 32000-2:2020 (PDF 2.0), has consolidated the format as a universal documentary infrastructure. From this foundation derive specialised profiles, PDF/A, PDF/X, PDF/E, PDF/UA, and PDF/VT, each designed to ensure compliance, interoperability, and reliability within specific domains such as archiving, printing, engineering, and accessibility.

The analysis extends to the software technologies that enable the generation, manipulation, and validation of interactive PDFs: in particular, the Apache PDFBox, iText 7, and OpenPDF libraries, as well as validation tools such as VeraPDF and PAC 2021. Through a comparative assessment of these tools, the study identifies the most suitable solutions for integration into automated workflows and digital document intelligence processes.

From an application point of view, the main use cases of interactive PDFs in the educational, healthcare, administrative, and industrial domains are illustrated here and demonstrate how the format can serve as an active interface between the user and the information system. The standard also presents best design practices, accessibility criteria compliant with PDF/UA and WCAG 2.1, and a set of operational guidelines for the production and validation of accessible and sustainable interactive documents.

This section highlights the emerging role of PDF as a platform for document intelligence, capable of integrating artificial intelligence, semantic interoperability, and process automation. The interactive PDF thus appears no longer as a mere distribution format, but as an active component of the digital ecosystem, combining formal stability, accessibility, and informational dynamism

The Portable Document Format (PDF), introduced by Adobe Systems in 1993, was conceived to provide a universal means of representing and distributing digital documents independently of platform, hardware, and software. Its strength lies in visual fidelity: a PDF document preserves typographic appearance, fonts, images, and formatting, ensuring consistent rendering across screen and print. Over the course of more than three decades, PDF has moved beyond its original role as a static container to become a reference standard for the production, dissemination, and archiving of digital knowledge. With the publication of ISO 32000-1:2008 and the subsequent ISO 32000-2:2020 (PDF 2.0), the format has been formally recognized as an international standard, placing it at the center of increasingly complex and automated document ecosystems.

The evolution of the PDF format has coincided with the growing need for documents that can interact with the user and integrate into digital information flows. Starting with the introduction of AcroForm modules, PDF incorporated direct input mechanisms (text fields, buttons, checkboxes, drop-down menus), paving the way for interactive PDFs: “living” documents that can respond to events, perform calculations, validate data, and integrate multimedia content.

The inclusion of JavaScript within the PDF model further extended these capabilities, allowing the definition of dynamic behavioural logic and application-level automation. Today, a PDF is no longer merely a representation of content, but an interactive environment capable of integrating elements of computation, communication, and control.

PDF differs from other formats through its object-oriented architecture, which ensures modularity, extensibility, and persistence. These characteristics make it the technological core of multiple derivative standards, each oriented towards a specific domain:

- PDF/A (ISO 19005) for long-term archiving;
- PDF/X (ISO 15930) for professional printing;
- PDF/E (ISO 24517) for engineering documentation;
- PDF/UA (ISO 14289) for universal accessibility;
- PDF/VT (ISO 16612-2) for variable data printing.

Each of these profiles introduces constraints and compliance requirements that ensure reproducibility and readability over time. In today’s digital ecosystem, PDF therefore represents a universal document platform, adaptable to diverse needs, ranging from administrative management to scientific research, from education to Industry 4.0.

The progressive digitisation of processes has led to the replacement of paper with smart forms and interactive formats capable of reducing time, errors, and operating costs. In this context, the interactive PDF acts as a bridge between document and application, enabling:

- data capture and automatic validation directly within the document;
- integration with information systems and web platforms;
- personalisation of content based on the user profile;
- engaging educational and multimedia experiences.

The application domains are manifold: education and training, with self-assessment tests and interactive manuals; healthcare, for managing clinical forms and anamnesis modules; public administration, for simplifying digital procedures; and industry, for updatable technical manuals and process documentation.

Objectives and structure of the report. This report aims to provide an organised and systematic view of the PDF format, with particular attention to the technical, regulatory, and application aspects that enable its use as an interactive and interoperable tool. The main objectives are to analyse the internal structure and normative evolution of the format; describe the derivative standards (PDF/A, PDF/UA, etc.) and their purposes; explore the techniques for creating interactive PDFs, with a focus on scripting and multimedia integration; present the technologies and software libraries (e.g., Apache PDFBox, iText, VeraPDF) that support generation and validation; define testing methodologies and conformity checks against ISO standards; and provide operational guidelines for designing and producing sustainable, accessible, and interoperable interactive PDFs.

The structure of the document reflects this articulation: Chapter 2 introduces the technical structure and PDF standards; Chapter 3 delves into the concept of interactivity and the main use cases; Chapters 4–8 examine implementation aspects, generation technologies, and validation frameworks; Chapter 9 presents testing and quality-control strategies; Chapter 10 offers conclusions and future perspectives; finally, an Appendix provides practical guidelines for producing compliant interactive PDFs.

The Portable Document Format (PDF), is today the de facto standard for the representation and distribution of digital documents. Its widespread adoption stems from its ability to preserve the visual appearance of a document regardless of the operating system, hardware, or viewing software used. Beyond guaranteeing portability and stability, PDF has evolved into a sophisticated container capable of integrating text, raster and vector graphics, multimedia content, annotations, structured metadata, and interactive scripts. Although invisible to the end user, its internal architecture reflects a highly modular design philosophy: every element, from a simple block of text to a 3D object, is represented as a uniquely identified structured object. This model, originally defined by Adobe and later formalised by the International Organization for Standardization (ISO), constitutes the foundation of interoperability that allows PDF to be treated as a document description language, analogous in some respects to a binary markup system.

A PDF document is organised into four main sections:

1. Header – contains the declaration of the PDF version (e.g., %PDF-1.7) and defines compatibility with software parsers.

2. Body – the core of the document, composed of a collection of numbered objects (text, images, vector graphics, annotations, fonts, patterns, metadata). Each object is described by a dictionary of attributes, specifying type, resources, and relationships with other objects.

3. Cross-Reference Table (XRef) – an indexed structure that maps the physical position of each object in the file. This table enables efficient random access to objects and supports incremental updates without requiring the entire file to be rewritten.

4. Trailer – contains global information about the document, such as the root object (catalogue), the total number of objects, and the offset of the XRef table.

The object-oriented model of PDF enables the selective manipulation of specific elements (for example, updating a metadata entry or replacing an image) without requiring the entire file to be recreated.

Its hierarchical structure enables modular extensions and backward compatibility, allowing a modern viewer to interpret PDF files produced decades ago.

In 2008, Adobe transferred the ownership and maintenance of the PDF format to ISO, marking the birth of ISO 32000-1:2008, which formalised the technical specifications of PDF 1.7. This transition ensured openness, transparency, and interoperability of the format, encouraging the development of software independent of the original vendor.

The subsequent revision, ISO 32000-2:2020, known as PDF 2.0, introduced significant improvements:

- a more coherent metadata management system based on XMP (Extensible Metadata Platform);
- a new graphic transparency model, simplifying the composition of visual layers;
- a stricter definition of annotations and digital signatures to enhance document security and authenticity;
- the introduction of Associated Files, enabling the linkage of external resources (e.g., XML, CSV, CAD models) while preserving semantic coherence with the main document;
- improved handling of multilingual content and embeddable fonts.

With PDF 2.0, the format has reached a technical maturity that allows its use as a universal infrastructure for interactive documents, digital archives, and e-government applications.

Every element within a PDF is an object, identified by a pair of numbers (*Object Number and Generation Number*) and defined using a syntax such as:

```
12 0 obj
<< /Type /Page
   /Parent 3 0 R
   /Contents 15 0 R
   /Resources 20 0 R >>
endobj
```

The /Type dictionary defines the nature of the object, while references (e.g., 15 0 R) link associated resources. This mechanism allows for efficient memory management and dynamic document manipulation.

PDF objects can belong to various types: boolean, numeric, string, name, array, dictionary, or stream. Streams contain compressed binary data such as images, fonts, or graphic content. The combination of logical and physical objects defines the layout and semantics of the document.

The universality of PDF has led to the creation of derived standards, each targeting a specific use case. Although based on the same ISO 32000 core, these standards impose additional constraints and rules to ensure compliance in professional contexts.

Table 1: Main Derived PDF Standards and Their Application Domains

Standard	ISO Code	Primary Objective	Domain of Use
PDF/A	ISO 19005	Long-term archiving	Digital archives, libraries, public administration
PDF/X	ISO 15930	Reliable exchange for printing	Publishing and typography
PDF/E	ISO 24517	Engineering and CAD documentation	Engineering, AEC

PDF/UA	ISO 14289	Universal accessibility	Documents for all users
PDF/VT	ISO 16612-2	Variable and personalised printing	Marketing, direct mailing

Each of these profiles defines a subset of rules:

- PDF/A prohibits encryption and requires all fonts to be embedded;
- PDF/UA enforces semantic tagging to improve compatibility with screen readers;
- PDF/X specifies strict limits on colour spaces, transparency, and output formats to ensure print fidelity.

The PDF/A standard (ISO 19005) is designed to ensure the readability of documents over the long term, regardless of future software or hardware.

To achieve this goal, it imposes strict constraints:

- all fonts must be embedded within the file;
- external references (e.g., web links or databases) are prohibited;
- encryption is not permitted;
- ICC colour profiles must be explicitly declared.

The main variants are:

- PDF/A-1 (2005) – based on PDF 1.4, backward compatible with Acrobat 5;
- PDF/A-2 (2011) – adds support for transparency, layers, and JPEG2000;
- PDF/A-3 (2012) – allows embedding of files of any type (e.g., XML, CSV, TXT) while maintaining compliance.

The latter version is particularly useful in hybrid archiving workflows, where the PDF document is associated with the original structured data, facilitating subsequent automated processing.

The PDF/UA (Universal Accessibility) standard, published in 2012 as ISO 14289-1, defines the requirements for making PDF documents accessible to users with disabilities.

Its main elements include:

- the presence of semantic tags (Heading, Paragraph, Table, List, etc.);
- a logical reading order consistent with visual layout;
- the use of alternative text for images and graphics;
- the correct identification of form fields and annotations.

A PDF compliant with PDF/UA can be correctly interpreted by assistive technologies such as screen readers or Braille displays, and adheres to the principles of Design for All and digital inclusion.

The PDF 2.0 specification (ISO 32000-2:2020) is not merely a technical update but a complete re-foundation of the PDF model. It introduces:

- a more coherent and rigorous language, free from undocumented dependencies on Adobe Acrobat;
- improved security management with AES 256-bit encryption;
- support for multiple digital signatures and PAdES-compliant timestamps;
- compatibility with emerging standards for semantic metadata (RDF, XML, JSON-LD);

- the explicit definition of document part metadata, enabling the structuring of complex, multi-section or versioned documents.

The aim of PDF 2.0 is to foster semantic interoperability, allowing documents to be not only viewed but also understood by intelligent systems and integrated into automated information flows. The evolution of the PDF format represents a paradigmatic example of successful technological standardisation. From a proprietary format to a global infrastructure, it embodies the synthesis of portability, stability, and adaptability. The derived standards, PDF/A, PDF/UA, PDF/X, and PDF/E, extend its applicability, ensuring compliance and sustainability over time.

3.5 Interactive PDFs

The progressive digitalisation of document workflows has transformed the PDF format from a simple tool for static representation into a platform for interactive communication.
While the first generation of PDFs in the 1990s was intended exclusively for viewing and printing, the evolution of ISO specifications and the introduction of dynamic APIs have enabled the emergence of interactive PDFs, documents capable of responding to user input and integrating application logic.

Interactivity has represented a turning point: the document is no longer merely the final outcome of a process, but becomes a functional interface in its own right, able to collect data, trigger procedures, and provide immediate feedback.

Interactivity in PDFs developed gradually through four main stages:

1. Static PDF (1990s) – a format oriented towards printing and faithful layout representation, with no active components.
2. Annotated PDF – introduction of annotations (comments, notes, links), which added an initial level of interaction.
3. Interactive Forms (AcroForms) – introduction of form fields and user controls such as text boxes, buttons, and drop-down menus.
4. Dynamic PDF (JavaScript and Multimedia) – inclusion of event handling, calculations, and dynamic content (audio, video, 3D).

This evolution, initiated with Acrobat 3 and consolidated with the PDF 1.5 specification, has made the format a vehicle for distributed applications, usable even in offline contexts. An interactive document should respond to a user action through programmed behaviours or event-driven logic. It can take several forms:

- Functional interactivity: filling out, validating, and submitting forms;
- Hypertextual interactivity: internal or external links, navigable indexes, anchors;
- Multimedia interactivity: embedding of audio, video, animations, or 3D models;
- Semantic interactivity: adaptation of content according to the user profile or document metadata.

These functions, enabled by components such as form fields, annotations, scripts, and action dictionaries, transform PDF into a genuine application platform, while maintaining compatibility with traditional viewing tools. Creating an interactive PDF requires combining multiple content layers.

Table 2: Interactive Components of PDF Documents and Related Use Examples

Component	Description	Practical Example
Form Field (AcroForm)	Interactive field defined in the document's /AcroForm dictionary	Text box for name, checkbox for consent
Annotation	Interactive object associated with a portion of a page	Comment, note, highlight, link
Action	Operation triggered by an event	Opening a URL, submitting a form, executing JavaScript
Multimedia Annotation	RichMedia element containing audio/video or 3D content	Embedded tutorial video in a manual
JavaScript Dictionary	Script associated with interface events	Validation of numeric or email fields

All these components are registered in the document catalogue and form part of the logical structure defined by the PDF object model.

One of the most significant aspects of interactive PDFs is the integration of JavaScript, which allows the definition of dynamic behaviours. Functions can be executed in response to events such as *MouseUp*, *Focus*, *Keystroke*, or *Validate*. An example of numeric field validation is reported in the following:

```
if (event.value < 0 || event.value > 100) {
app.alert("The value must be between 0 and 100.");
event.rc = false;
}
```

Thanks to this capability, the PDF becomes a controlled execution environment, able to react to user interaction while maintaining security through the sandboxing mechanisms implemented in modern PDF viewers.

Designing accessible interactive PDFs requires attention to digital inclusion principles. The PDF/UA standard (ISO 14289) establishes specific requirements, including:

- use of semantic tags consistent with the logical structure of the document;
- definition of the correct reading order for screen readers;
- labelling of form fields with meaningful names;
- inclusion of alternative text (alt text) for images and multimedia content;
- compliance with WCAG 2.1 and the European Directive EN 301 549.

An accessible interactive PDF not only ensures usability for all users but also enhances indexing and content search, improving interoperability with automated analysis and artificial intelligence tools.

The main characteristics that distinguish an interactive PDF from a static one can be summarised as follows:

1. Modularity – ability to integrate text, graphics, data, audio, and video within a single file;
2. Personalisation – possibility to adapt content and fields according to user profile or metadata;
3. Data Collection – support for fillable forms exportable to XML or JSON;

4. Responsiveness – real-time reaction to user inputs (calculations, validations, messages);
5. Connectivity – interaction with web systems or external databases via HTTP or JavaScript;
6. Compatibility – consistent display across devices and platforms.

These features make the interactive PDF a versatile medium for document automation and bidirectional communication scenarios. Interactive PDFs are now widely used across multiple sectors. In all these contexts, interactivity turns the document into an intelligent contact point between user and information system, empowering the users performing their activities.

Table 3: Main Application Domains of Interactive PDFs and Related Functional Benefits

Sector	Typical Application	Main Benefit
Education	Self-assessment tests, interactive handouts, manuals with embedded videos	Personalised and dynamic learning
Healthcare	Fillable anamnesis forms integrated with HIS systems	Error reduction and process digitalisation
Public Administration	Digital forms for applications and self-certifications	Efficiency and reduction of paper bureaucracy
Industry	Technical manuals and interactive catalogues	Rapid updating and multimedia access
Marketing and Communication	Interactive brochures and multimedia catalogues	Engagement and interaction tracking

The interactive PDF presents some challenges that must be carefully managed:

- Compatibility: not all viewers fully support JavaScript or multimedia content;
- Security: script execution can pose risks if not properly restricted;
- Performance: highly complex files may be heavy or slow on mobile devices;
- Archiving: certain dynamic features are not permitted in PDF/A profiles, making a dual-format approach (interactive + archivable) necessary.

Effective design therefore, requires a balance between interactivity and stability, ensuring both accessibility and document longevity.

The interactive PDF represents one of the most significant evolutions in digital documentation. Thanks to its ability to combine formal stability, universal compatibility, and dynamic functionality, PDF has become a hybrid platform between document and application. Its applications range from education to public administration, healthcare, and industry, confirming it as a key tool for digital transformation. When designed with attention to clarity and accessibility, interactivity not only enhances user experience but also consolidates PDF as a living, sustainable format within the contemporary documentary ecosystem.

3.6 Libraries and Technologies

This section provides an examination of the most widely used libraries and technologies for producing and manipulating interactive PDFs, with a particular focus on aspects of interoperability, accessibility, security, and ISO compliance. The analysis integrates architectural perspectives (modules, object models, resource management), operational perspectives (APIs, integration pipelines), and regulatory considerations (PDF/A, PDF/UA) to provide solid selection criteria for academic, industrial, and administrative contexts.

PDF libraries expose an object model that maps the fundamental elements of the format: documents, pages, resources, streams, annotations, metadata, and tagging structures. Parsing involves reading the XRef table and the trailer, reconstructing indirect objects and resolving cross-references. Rendering requires interpreting the graphic instruction set (based on PostScript/PDF) and managing fonts, transparency, and colour profiles. The quality of the parsing engine directly affects both the rendering fidelity and the ability to perform non-destructive editing (incremental update).

Interactive forms are historically based on AcroForm (fields, widgets, appearance dictionaries) and, in some legacy ecosystems, on XFA. JavaScript scripting enables client-side validation, calculations, and event-driven reactions, but introduces compatibility constraints across PDF readers. Best practices recommend cautious use of scripts and a design approach that prioritises standard components, in order to maximise portability and preserve security. The use of tagged structures (required for PDF/UA compliance) allows interactivity that remains compatible with assistive technologies.

XMP metadata (embedded XML) allows precise description of content, rights, and relationships, facilitating indexing and long-term preservation. Compliance with PDF/UA requires the use of semantic tags, consistent reading order, alternative text for images, and navigable structure. Font embedding is essential to ensure typographic fidelity and platform independence; the absence of embedded fonts may violate PDF/A constraints.

A structured comparison of the most commonly used libraries for interactive PDFs in academic and industrial contexts are listed in the following:

Table 4: Comparative Overview of Libraries for Interactive PDFs (Architecture and Features)

Library	Language	Licence	Creation	Editing	Forms/JS	Compliance
Apache PDFBox	Java	Apache 2.0	Yes	Yes	Forms (limited JS)	Preflight (PDF/A)
iText 7	Java/.NET	AGPL / Commercial	Yes	Yes	Forms + JS	Extensive support
OpenPDF	Java	LGPL/MPL	Yes	Yes	Forms	Partial
ReportLab	Python	BSD	Yes	Limited	Custom	External
PDF.js	JavaScript	Apache 2.0	No (render only)	No	N/A	N/A

Apache PDFBox is a mature Apache project oriented toward Java, featuring core modules (creation/modification), FontBox (font management), XMPBox (metadata), and Preflight (PDF/A validation). It is suitable for server-side pipelines and batch processing. Strengths: permissive licence, JVM integration, good support for forms and metadata, PDF/A validation. Limitations: limited support for JavaScript and advanced multimedia.

iText 7 provides rich APIs for advanced document creation, forms, digital signing, and complex layouts. It is distributed under an AGPL or commercial licence. Strengths: performance, security, form workflow and interactivity, comprehensive documentation. Considerations: licensing regime and costs for commercial use; careful management of compatibility across heterogeneous readers.

OpenPDF is an open-source fork (LGPL/MPL) maintaining partial compatibility with the historical iText 2.x API. It is suitable for projects requiring copyleft licences or compatibility with existing codebases. Advanced features and full standards compliance may require additional integration.

ReportLab (Python) is effective for programmatic generation and templating, excelling in report layout construction. Editing existing PDFs is more limited. Its integration with the Python ecosystem (pandas, NumPy) facilitates the creation of dynamic reports.

PDF.js is a browser-based rendering engine, useful for client-side viewing and annotation. It is not designed for PDF generation or deep modification, but for web integration and interactive display.

Performance depends on resource caching (fonts, images), parallelisation (batch operations), and I/O minimisation. Compatibility varies due to differences among PDF readers (Acrobat, Okular, browser engines). Interactive functions based on JavaScript are not uniformly supported. For high-volume scenarios, it is advisable to separate pre-processing, generation, and validation phases, with structured metrics and logging.

Digital signatures, based on X.509 certificates and PKCS#7/CAdES envelopes, ensure integrity and authenticity. Timestamping (TSA, RFC 3161) provides proof of existence. PDF/A compliance imposes constraints on fonts, colour management, transparency, and dynamic content. Preflight tools (PDFBox) or iText modules allow both validation and, in some cases, automated correction.

The choice of library depends on functional requirements (types of interactivity, scripting), non-functional aspects (performance, scalability), licence constraints, and integration with the existing ecosystem. In enterprise contexts, the availability of commercial support and auxiliary tooling (e.g., validation) can be a decisive factor. Accessibility requirements demand particular attention to PDF/UA compliance and proper tagging.

For the creation of robust and interoperable interactive PDFs, the following practices are recommended:

1. Adoption of relevant standards (PDF/A, PDF/UA);
2. Limiting script use to essential cases;
3. Complete embedding of fonts;
4. Automated Preflight validation;
5. Quality metrics and logging;
6. Separation of template and data layers.

Library selection should balance cost, features, and long-term maintainability.

The analysis of the internal architecture of interactive PDFs highlights the complexity and modularity of the format, which can be effectively leveraged only through advanced manipulation tools. The following chapter will further explore the software libraries and development environments that enable the generation, modification, and automation of interactive PDFs, with particular focus on open-source Java-based solutions.

3.7 Java and the Programmatic Management

This chapter provides an in-depth analysis of how the Java language supports the creation, manipulation, and validation of PDFs, with particular emphasis on interactive documents that include forms, scripting, and advanced metadata. The discussion covers the main architectural patterns, the APIs of key libraries (Apache PDFBox, iText 7, OpenPDF), integration strategies for server-side pipelines, and best practices for performance, security, and standards compliance. The goal is to offer practical guidelines for designing scalable, maintainable, and ISO-compliant (PDF/A, PDF/UA) solutions.

An industrial Java-based document pipeline typically includes:

- input acquisition and normalisation (CSV, JSON, XML);
- schema validation;
- data mapping to templates or low-level APIs;
- PDF generation;
- compliance verification (e.g., PDF/A with Preflight);
- optional digital signing and timestamping;
- distribution and archiving.
-

The use of message queues (e.g., JMS/Kafka) and REST services facilitates scalability and operational observability.

The use of Factory and Builder patterns simplifies the composition of complex documents, while the Facade pattern encapsulates the complexity of PDF APIs. It is recommended to use try-with-resources for safe stream management and DTO/mapper layers to separate data domain from layout logic. Structured logging (correlations, library versions, file checksums) enables auditability and traceability.

Idiomatic examples and a comparative overview of three widely adopted libraries are listed in the following:

Tabella 5: – Comparative Overview of Java Libraries for PDFs

Library	Strengths	Limitations	Licence	Notes
Apache PDFBox	Mature open-source project; Preflight; JVM integration	Limited JS support; reduced advanced multimedia	Apache 2.0	Excellent for server batch processing
iText 7	Rich APIs; signatures; advanced layouts; high performance	AGPL/Commercial licence	AGPL/Commercial	Enterprise-grade solution
OpenPDF	Legacy compatibility; copyleft model	Partial support for advanced features	LGPL/MPL	Good OSS compromise

Creating a PDF with Apache PDFBox (basic document, one page):

```
try (PDDocument doc = new PDDocument()) {
PDPage page = new PDPage();
doc.addPage(page);
try (PDPageContentStream cs = new PDPageContentStream(doc, page)) {
        cs.beginText();
        cs.setFont(PDType1Font.HELVETICA_BOLD, 14);
        cs.newLineAtOffset(72, 720);
        cs.showText("Hello PDFBox");
        cs.endText();
}
doc.save("output.pdf");
}
```

Filling a form field with iText 7:

```
PdfDocument pdfDoc = new PdfDocument(new PdfReader(src), new PdfWriter(dest));
PdfAcroForm form = PdfAcroForm.getAcroForm(pdfDoc, true);
PdfFormField field = form.getField("name");
field.setValue("John Smith");
form.flattenFields();
pdfDoc.close();
```

Digital signature with OpenPDF (simplified schema):

```
PdfReader reader = new PdfReader(src);
PdfStamper stamper = PdfStamper.createSignature(reader, new FileOutputStream(dest),
'\0');
PrivateKey pk = ...; Certificate[] chain = ...;
ExternalDigest digest = new BouncyCastleDigest();
ExternalSignature signature = new PrivateKeySignature(pk, "SHA256withRSA", "BC");
MakeSignature.signDetached(stamper.getSignatureAppearance(), digest, signature, chain,
null, null, null, 0, MakeSignature.CryptoStandard.CMS);
```

XMP metadata describes content, rights, and relationships, and are essential for indexing and preservation. Interactive forms (AcroForms) require clear labels, formats, and constraints; JavaScript usage should remain cautious to ensure compatibility. Digital signatures based on X.509/PKCS#7/CAdES ensure integrity and authenticity, while TSA timestamping (RFC 3161) provides proof of existence.

Performance improves through font/image caching, reduced I/O, batch parallelism, and the use of templates. Security involves AES-256 encryption, permission handling, path hardening, and input sanitisation. Compatibility across PDF readers is not uniform for JavaScript and multimedia content, requiring testing on Acrobat, browser engines, and open-source viewers.

Table 6: Tools, Libraries, and Validators for the Generation and Verification of Interactive PDFs

Tool	Category	Brief Description	Main Features	Typical Use Case
Apache PDFBox	Open-source	Java library for complete PDF management	Creation, manipulation, digital signing, PDF/A validation	Java application development, CI/CD integration
iText 7 (AGPL)	Open-source / Commercial	Cross-platform toolkit for dynamic PDF creation	Document generation, interactive forms, digital signatures, watermarking, templating	Enterprise applications and document automation
OpenPDF	Open-source	Independent fork of iText 4	Creation and editing of PDFs, form and annotation handling	Lightweight OSS solutions

VeraPDF	Open-source	Official ISO PDF/A validator	PDF/A compliance validation, XML/JSON reports, CLI and Java API	Digital archives and public administration audits
Apache PDFBox Preflight	Open-source	Integrated validation module within PDFBox	Programmatic PDF/A checking, CI pipeline integration	Automated testing in development environments
PDF Accessibility Checker (PAC 2021)	Open-source	Official tool for PDF/UA verification	Accessibility analysis, tagging and reading order tests	Accessibility and inclusion compliance
Adobe Acrobat Preflight	Commercial	Advanced verification module in Acrobat Pro	PDF/A/X/UA validation, automatic correction, custom profiles	Publishing, printing, public institutions
Ghostscript / Ghostscript Validator	Open-source	PostScript engine and PDF converter	PS↔PDF conversion, compression, compatibility check, batch processing	Conversion automation and file optimisation
LaTeX + media9	Open-source	Package for embedding multimedia in PDFs	Integration of videos, 3D animations, and RichMedia content	Scientific and technical documentation
PDFUnit	Java Library	Framework for PDF unit testing	Automated verification of layout, text, links, and metadata	Software QA and automated testing
PAC CLI Toolkit	Open-source	Command-line interface for PAC 2021	Batch PDF/UA validation, structured report export	Integration into automated workflows
GhostPDFA	Open-source	Ghostscript extension for PDF/A	Conversion and compliance checking for PDF/A	Document archiving and preservation

In order to designing Java document pipelines, it is recommended to:

1. strictly separate templates and data;
2. apply input validation with explicit schemas;
3. use Factory/Builder/Facade patterns to manage complexity;
4. automate Preflight and PDF/UA compliance checks;
5. integrate digital signatures and timestamping according to organisational policies;
6. monitor key performance metrics (execution time, file size, errors);
7. include cross-viewer and regression testing;
8. document library versions and dependencies for reproducibility.

The discussion of major processing libraries, Apache PDFBox, iText, and OpenPDF, has shown that it is now possible to generate highly customised and technically compliant interactive PDFs. However, the creation of complex documents introduces the challenge of verifying compliance with international standards and validating interactive content, which are fundamental for ensuring interoperability and durability. The next chapter therefore, systematically addresses the topic of compliance and validation, analysing relevant ISO standards and dedicated tools such as VeraPDF and PAC 2021.

3.8 Standards, Compliance, and Validation

Over the past two decades, the PDF (Portable Document Format) has evolved from a simple static container of graphical and textual information into a complex infrastructure for the representation, exchange, and preservation of digital content. Interactivity, introduced through the integration of forms, scripts, hyperlinks, and multimedia content, has greatly expanded the functional scope of the format, but has also made the definition of compliance and validation rules increasingly necessary.

The concept of compliance refers to the ability of a PDF document to meet a codified set of technical and semantic requirements, as defined by ISO specifications or organisational policies. Validation, on the other hand, is the process of verifying whether a document actually satisfies these requirements.

This chapter provides an in-depth discussion of the ISO standards underlying the PDF format, the specific profile variants (PDF/A, PDF/UA, PDF/X, PDF/E), the tools available for automated validation, and the methodological criteria for integrating these checks within industrial and academic pipelines.

Originally developed by Adobe Systems, the PDF format was standardised by ISO in 2008 with the publication of ISO 32000-1:2008, which established its openness and independence from proprietary implementations. In 2020, ISO 32000-2:2020 (known as PDF 2.0) redefined the data model, introducing a more consistent approach to metadata, annotations, and multimedia management.

PDF 2.0 improved colour transparency, font handling, cryptographic security, and the semantic representation of content. It also introduced the concept of “Associated Files”, allowing the structured and unified embedding of external resources. From the ISO 32000 corpus derive several profiled standards, each with specific objectives:

Table 7: Comparison of ISO Standards for the PDF Format: Objectives, Constraints, and Application Domains

Standard	Main Objective	Main Constraints	Application Domain
PDF/A	Long-term archiving	No encryption, embedded fonts, mandatory ICC colour profiles	Digital archives, libraries, public administration
PDF/UA	Universal accessibility	Semantic tagging, reading order, alternative text for images	Public administration, education
PDF/X	Reliable exchange for printing	Controlled colour spaces, exclusion of unmanaged transparencies	Printing industry, publishing
PDF/E	Technical and engineering exchange	Support for 3D models and CAD attachments	Engineering, AEC, mechanical design

The variety of standards reflects the need to ensure interoperability and compliance across heterogeneous application domains, reducing risks of information loss and ensuring long-term reproducibility. The adoption of standardised profiles ensures three fundamental properties:

- Reliability – A compliant document can be read and interpreted consistently over time and across systems.
- Accessibility – PDF/UA documents guarantee content usability for users with disabilities.
- Digital Preservation – PDF/A profiles ensure independence from external resources and informational self-sufficiency.

In industrial or academic production contexts, these properties represent not only technical requirements, but also quality indicators for the document lifecycle, often subject to audit or certification (e.g., ISO 9001 or ISO 14721 OAIS for archiving).

Compliance validation is a process that can be automated. The most widely used tools can be grouped into two main categories:

- Commercial tools (e.g., Adobe Acrobat Preflight), offering advanced graphical interfaces, detailed reporting, and multi-platform support.
- Open-source tools, such as VeraPDF or Apache PDFBox Preflight, which can be integrated into automated pipelines and server systems.

Tabella 8: Operational Comparison of Validation Tools

Tool	Type	Standard Support	Operational Notes
Adobe Acrobat Preflight	Commercial	PDF/A, PDF/X, PDF/UA	Full GUI, detailed reporting
VeraPDF	Open-source	PDF/A	CLI and Java API, adopted by European national archives
Apache PDFBox Preflight	Open-source	PDF/A	Integrable in Java pipelines, excellent for server-side automation
Ghostscript	Open-source	PDF/A	Batch conversions, CLI scripting

The choice of tool depends on the operational context: Adobe Preflight is ideal for editorial environments, while PDFBox Preflight and VeraPDF are best suited for automated, open-source workflows.

The Preflight module of Apache PDFBox enables programmatic PDF/A compliance checking. An example in Java follows:

```
PreflightParser parser = new PreflightParser(new File("document.pdf"));
parser.parse();
try (PreflightDocument document = parser.getPreflightDocument()) {
document.validate();
ValidationResult result = document.getResult();
if (result.isValid()) {
        System.out.println("Document is PDF/A compliant");
} else {
        result.getErrorsList().forEach(e -> System.out.println(e.getDetails()));
}
}
```

The previous snippet can be integrated into a document generation system or REST service to automatically validate inbound or outbound PDFs.

VeraPDF, developed by the Open Preservation Foundation with support from the European Commission, is the de facto standard for open-source PDF/A validation.
Typical terminal execution:

```
verapdf --format text --policy PDFA-1B document.pdf
```

The output summarises the validation results, indicating compliance status and possible semantic errors. The latest version supports REST API invocation and XML report generation.

Automatic validation is a core component of document automation pipelines, especially in the context of large-scale interactive document production. Compliance checks can be integrated at various stages of the document lifecycle:

1. Generation stage – Immediate validation after PDF creation to detect discrepancies early.
2. Publication stage – Quality control before distribution or archiving.
3. Preservation stage – Periodic verification of readability and file integrity.

Automation can be implemented via CI/CD systems (e.g., Jenkins, GitLab CI), with tasks invoking PDFBox Preflight or VeraPDF CLI. Results can be stored in document databases or structured logs, enabling statistical analysis of compliance trends over time.

The most significant evolutions in PDF standardisation include:

- PDF/UA-2 (ISO adoption expected in 2025), introducing a more rigorous semantic model and enhanced interoperability with accessible web formats (HTML5, EPUB3).
- PDF 2.0 (ISO 32000-2), providing a unified foundation for future ISO profiles and improving coherence across PDF/A, PDF/UA, and PDF/X.
- Integration with RDF/OWL structured metadata models, enabling reuse and semantic search of content.

The adoption of semantic metadata will be crucial for ensuring interoperability between document repositories, content management systems, and AI-driven platforms for automated document analysis.

Standardising interactive PDFs still presents several open challenges:

1. Dynamic content management – PDFs containing scripts, videos, or 3D components require extended specifications to ensure security and reproducibility.
2. Semantic validation – Current standards focus on technical structure but not yet on semantic coherence (e.g., logical relations among interactive forms).
3. AI interoperability – Automated analysis systems must be able to interpret PDF/UA and PDF/A metadata to provide accurate semantic feedback.
4. Digital preservation – The sustainability of PDF as a long-term archival format depends on maintaining accessibility of embedded multimedia resources.

The future of interactive PDFs will thus be defined by a balance between technological evolution and regulatory continuity. Joint efforts by ISO committees, open-source foundations (such as VeraPDF and the PDF Association), and academic developers aim to build a more transparent, validatable, and AI-integrable document ecosystem.

After examining the reference standards and validation criteria, the following chapter explores advanced interactivity and personalisation techniques, which push the PDF format beyond its traditional document function toward intelligent, adaptive, and context-aware content.

3.9 Advanced Techniques for Interactivity and Personalisation

The evolution of interactive PDFs has transformed the format from a static information container into a dynamic platform capable of supporting complex, customisable user experiences, fully integrated with web applications and cloud services. Today, PDF documents can incorporate fillable forms, scripting logic, multimedia content, connections to external databases, and even 3D components, paving the way for new forms of interaction and data analysis. In academic, corporate, and administrative contexts, these capabilities enable intelligent workflows: the document is no longer just a final output, but becomes an active element of the information process, capable of collecting data, responding to user input, and adapting to its operational context.

This chapter examines the main advanced techniques for developing interactive PDFs, analysing their technical mechanisms, available software tools, and design best practices.

PDF forms (form fields) represent the most common form of interactivity.
The PDF standard supports various field types:

- Text fields (TextField) – allow users to input free text;
- Check boxes (CheckBox) – enable binary selections (yes/no);
- Radio buttons (RadioButton) – define mutually exclusive option groups;
- Drop-down menus and list boxes (ComboBox/ListBox) – allow selection from predefined lists;
- Action buttons (PushButton) – can submit forms, execute scripts, or open external links.

An interactive document may contain hundreds of coordinated fields, logically organised through hierarchies and group names. In Java environments, libraries such as iText 7 and Apache PDFBox allow programmatic creation of PDF forms.

Example (simplified, using iText 7):

```
PdfDocument pdf = new PdfDocument(new PdfWriter("form.pdf"));
Document doc = new Document(pdf);
PdfAcroForm form = PdfAcroForm.getAcroForm(pdf, true);
PdfTextFormField name = PdfTextFormField.createText(pdf, new Rectangle(100, 700, 200,
20), "name", "");
form.addField(name);
doc.close();
```

This code generates an interactive text field placed on the page.
Each field can have custom actions associated with it, such as validation or automatic submission.

PDFs support embedded JavaScript, which can handle events (e.g., onFocus, onBlur, onClick), validate data, or dynamically modify content.

Typical example:

```
if (this.getField("age").value < 18) {
app.alert("You must be at least 18 years old to complete this form.");
}
```

This capability allows the creation of self-validating, dynamic forms, but it also introduces potential security risks. For this reason, recent versions of Acrobat Reader and other viewers restrict the execution of potentially harmful scripts.

Modern applications adopt a hybrid approach: the PDF contains only the interactive structure, while the logic is executed by an external web service. This model is common in Document-as-a-Service systems, where PDFs act as interfaces for data flows processed by secure servers, reducing the risks associated with local scripting.

The PDF format supports direct integration of multimedia elements through RichMedia or Sound annotations.

Typical applications include:

- Technical manuals with embedded demonstration videos;
- Educational materials with audio narration;
- Marketing documents with interactive animations.

Multimedia embedding can be achieved using Adobe Acrobat Pro, LaTeX (media9), or dedicated APIs such as PDFBox RichMediaAnnotation. Ensuring codec compatibility (H.264, AAC) and local resource availability is essential.

PDF/E and PDF 2.0 standards support the display of 3D models via U3D or PRC formats. These models allow rotation, sectioning, and direct annotation. Engineering and industrial design applications use this capability to distribute interactive CAD models in a format accessible even to non-technical users.

PDFs can embed XMP metadata (Extensible Metadata Platform) to describe content, authorship, versioning, and rights. Such metadata can be leveraged to generate personalised document versions based on the user profile, using conditional rendering systems.

Example: an educational document could adapt its language, complexity, or level of detail based on the metadata of an authenticated user.

Frameworks such as Apache FOP, Docmosis, or iText Template Engine enable the creation of PDFs from XML or JSON templates, where dynamic variables are automatically substituted. This approach maintains a separation between content and presentation logic, improving maintainability and automation.

Interactivity also introduces new security challenges. PDF digital signatures (based on PKCS#7 and PAdES standards) ensure document authenticity and integrity.

The main signature types are:

1. Invisible signature (integrity) – applied without altering the visual layout;
2. Visible signature (identification) – includes a signature field with graphical representation;
3. Multiple signatures – support distributed approval workflows.

Open-source tools such as PDFBox and iText support the application and verification of PAdES signatures, integrating cryptographic modules such as BouncyCastle.

Example in Java with PDFBox:

```
PDDocument document = PDDocument.load(new File("contract.pdf"));
SignatureOptions options = new SignatureOptions();
PDSignature signature = new PDSignature();
signature.setName("Dr. Rossi");
signature.setReason("Technical approval");
document.addSignature(signature, new CreateSignature().signingInterface(), options);
document.save("contract_signed.pdf");
```

Interactive PDFs can interface with enterprise information systems through:

- HTTP Submit – sending form data to a web endpoint;
- FDF/XFDF – XML-based formats for transmitting filled form values;
- REST Webhooks – direct integration with server APIs (e.g., automatic submission of completed form data);
- RAG/AI Pipelines – semantic analysis of completed forms to extract knowledge or verify logical consistency.

These capabilities make the PDF an ideal intermediate component between user interfaces and backend systems.

The accessibility of interactive PDFs is governed by PDF/UA (ISO 14289) and, in Europe, by EN 301 549 and WCAG 2.1 directives. To ensure compliance, one must:

1. Structure the document with coherent semantic tags (headings, lists, tables);
2. Provide alternative text for images and form fields;
3. Define a logical reading order;
4. Test compatibility with assistive technologies (screen readers).

Tools such as PAC 2021 and Adobe Accessibility Checker support automatic validation and generation of accessibility compliance reports.

Current trends show a growing convergence between interactive PDFs and web technologies. Among the most significant developments:

- PDF as a Platform – PDFs as active components in SaaS ecosystems;
- Hybrid Forms – documents combining PDF and HTML5 functionalities;
- Semantic PDFs – documents enriched with RDF metadata for AI-based analysis;
- Augmented PDFs – integration of AR/VR for immersive 3D content visualisation.

The future of PDFs will increasingly focus on semantic transparency, interoperability, and security, aligned with the broader vision of document intelligence.

Interactivity in PDFs today represents one of the key innovation drivers in the field of document automation. Through scripting, form fields, dynamic personalisation, and integration with intelligent services, the PDF has evolved from a simple typographic output to a full-fledged application platform. The challenge for the coming years will be to balance expressive flexibility, accessibility, security, and digital sustainability, making the PDF a truly living medium within the information ecosystem.

The advanced design and personalisation strategies of interactive PDFs require clearly defined production workflows to be effective. The next chapter therefore describes best practices for managing document flows, from creation to distribution, ensuring quality, accessibility, and consistency throughout the entire document lifecycle.

3.10 Production Workflows

The production of interactive PDFs requires a structured methodological approach that integrates technical, organisational, and regulatory aspects. It is not merely a matter of “generating” a PDF file, but of designing a process that ensures quality, compliance, accessibility, and traceability throughout the entire document lifecycle.

A well-designed production workflow ensures:

- Content consistency, avoiding versioning or formatting errors;
- Operational efficiency, through the automation of repetitive tasks;
- Quality control, via automatic validation and review stages;
- Regulatory compliance, with respect to ISO standards, public administration guidelines, and digital archiving requirements.

An interactive PDF production workflow can be modelled as a modular pipeline, organised into six main macro-phases:

1. Design of the document's logical model
    a. Definition of content, logical structure, and expected interactivity;
    b. Identification of mandatory and optional metadata;
    c. Mapping of relationships between elements and external resources.
2. Automatic generation of the base document (template)
    a. Use of tools such as LaTeX, iText, Docmosis, or Apache FOP;
    b. Separation between content (XML/JSON) and layout (XSL-FO or CSS).
3. Dynamic data population
    a. Extraction of information from databases, online forms, or management systems;
    b. Automatic filling of form fields or insertion of dynamic placeholders.
4. Application of interactivity and scripting
    a. Addition of form fields, validations, scripts, or multimedia elements;
    b. Definition of associated actions (data submission, URL opening, internal navigation).
5. Validation and compliance checking
    a. Structural verification (ISO 32000-2);
    b. Accessibility testing (PDF/UA);
    c. Archival validation (PDF/A);
    d. Semantic checking using AI tools or business rules.
6. Distribution and preservation
    a. Publication in repositories or cloud platforms;
    b. Digital signature and timestamping (PAdES, XAdES);
    c. Archival storage following OAIS models or ECM (Enterprise Content Management) systems.

Open-source tools represent a strategic choice for research institutions and public administrations, as they ensure transparency and customisable integration.

Table 9: Tools and Libraries for Generating and Validating Interactive PDF Documents

Tool	Brief Description	Main Features
Apache PDFBox	Open-source Java library	Creation, manipulation, signing, and PDF/A validation
iText 7 (AGPL)	Toolkit for dynamic PDFs	Generation, interactive forms, signatures, watermarks, templates
VeraPDF	Official ISO PDF/A validator	CLI/API for compliance validation

Ghostscript	PostScript–PDF converter	Optimisation, compression, compatibility checking
LaTeX + media9	Typesetting environment	Production of scientific and multimedia PDFs

Commercial platforms such as Adobe Experience Manager Forms, Foxit PDF Editor SDK, or DocuSign offer direct integration with complex enterprise workflows, supporting user management, versioning, and signature tracking. However, they come with higher costs and lower flexibility compared to open-source solutions.

It is good practice to maintain a clear separation between:

- Content (text, data, metadata);
- Presentation (layout, styles, graphics);
- Logic (validation rules, scripts, conditional flows).

This separation allows each layer to be updated independently, promoting reusability and system adaptability.

Every generated PDF should be linked to a specific version of its template, source data, and metadata schema. Using version control systems (e.g., Git, SVN) enables traceability of changes and rollback in case of errors.

Automatic validation tools (e.g., VeraPDF, PDFBox Preflight) should be integrated directly into the workflow pipeline. These tools can be invoked by CI/CD scripts or dedicated microservices, returning JSON or XML outputs for subsequent analysis.

Integrating interactive PDFs with artificial intelligence systems enables:

- Automatic analysis of filled content;
- Recognition of patterns or logical errors in forms;
- AI-based feedback and correction suggestions;
- Semantic verification against reference models.

These functionalities can be implemented through RAG (Retrieval-Augmented Generation) architectures, where a Large Language Model (LLM) accesses a document repository to validate and comment on PDF content. Embedding RDF/XML and JSON-LD metadata within PDFs allows for semantic indexing and integration with Linked Open Data platforms, enhancing traceability and reusability in e-government and e-learning contexts.

Digital signatures should be centrally managed, with clear policies defining roles, algorithms (RSA, ECDSA), and key lifetimes. Each signed PDF must retain information on the certificate, trust chain, and timestamp. Long-term preservation requires files to comply with PDF/A-3 or later and for signatures to follow PAdES-LTV (Long-Term Validation) specifications. Archival systems should include periodic integrity checks and redundant backups.

To assess the quality of the document workflow, quantitative metrics can be defined.

Table 10: Quality and Performance Indicators for Evaluating Interactive PDFs

Indicator	Description	Formula or Unit of Measure
Average PDF/A compliance	Percentage of valid documents	(# valid PDFs / # total PDFs) × 100

Average validation time	Efficiency of automated checks	seconds/document
Average number of revisions	Template and data quality	revisions/cycle
Metadata coverage	Completeness of XMP fields	% of populated fields

These indicators allow continuous monitoring and optimisation of system performance over time.

The most advanced workflows adopt event-driven or microservice architectures, where each process phase is managed by an autonomous component communicating via queues or REST APIs.

Typical example: Data Extraction Service → PDF Generation Service → Validation Service → Digital Signature Service → Archival Service.

This approach enables scalability, resilience, and integration with cloud-native platforms such as Kubernetes, Docker, or AWS Lambda.

1. Design for compliance – adhere to PDF/A and PDF/UA standards from the start.
2. Automate all repeatable tasks – minimise manual errors.
3. Use consistent naming conventions – for fields, templates, and files.
4. Validate before publishing – include mandatory CI/CD checks.
5. Monitor and log errors – produce periodic compliance reports.
6. Plan migration strategies – toward new formats (e.g., PDF 2.0 or HTML5).

A well-structured document workflow drastically reduces errors, costs, and production time. In the context of digital transformation, the interactive PDF has become a key medium for data communication and management. The integration of automatic generation, validation, and AI technologies makes it possible to build intelligent, transparent, and sustainable document systems.

Once the production workflow has been defined, it becomes essential to verify the quality and reliability of interactive PDFs through testing procedures and automated control tools. Chapter 9 will analyse in detail the methods of testing, validation, and monitoring applicable to PDF documents, with practical examples and open-source tools for compliance verification.

3.11 Testing and Experimental Validation Tools

Testing interactive PDFs represents a crucial stage in the digital document lifecycle, as it allows for verifying functional correctness, standards compliance, and user experience consistency. In advanced automation contexts, testing does not concern only document rendering, but also the execution of scripts, form field management, semantic validation, and the robustness of embedded multimedia content.

Modern PDF production workflows include unit tests, integration tests, and automated compliance tests, often managed by CI/CD pipelines or dedicated validation environments. This is especially critical in high-reliability domains, such as public administration, digital archiving, healthcare, or education, where document errors can lead to data loss or legal non-compliance.

Testing techniques for interactive PDF documents can be classified into four main categories.

Table 11: Types of Tests for the Technical and Functional Validation of Interactive PDFs

Test Type	Description	Main Objectives
Structural Tests	Verification of file integrity, absence of corrupted objects, correct XRef and trailer structure	Ensure technical validity of the document
Functional Tests	Validation of the behaviour of interactive fields, buttons, links, scripts, and forms	Ensure correct interactivity
Compliance Tests	Checking against ISO standards (PDF/A, PDF/UA, PDF/X) using validation tools	Ensure regulatory alignment
Accessibility Tests	Analysis of screen reader compatibility, reading order, and semantic tagging	Ensure accessibility and inclusiveness

Each test type can be automated and integrated into PDF generation workflows, reducing manual intervention and improving the quality of the final product.

Modern PDF validation pipelines rely on both open-source and commercial tools, offering graphical interfaces (GUI), APIs, and CLI support for reproducible testing.

Table 12: Validation Tools for Automated Verification of Interactive PDFs

Tool	Category	Main Features
VeraPDF	Open-source	PDF/A validation, XML/JSON reports, CLI and Java API
Apache PDFBox Preflight	Open-source	Programmatic PDF/A checking, CI/CD integration
PDF Accessibility Checker (PAC 2021)	Open-source	PDF/UA analysis, tagging and reading order testing
Adobe Acrobat Preflight	Commercial	PDF/A/X/UA validation, customisable profiles
Ghostscript Validator	Open-source	Compatibility verification, batch conversion
PDFUnit	Java library	Automated unit tests on generated PDFs (layout, text, links)

The choice of tool depends on the required integration level: VeraPDF and PDFBox Preflight are ideal for automated pipelines, while PAC 2021 and Acrobat Preflight are more suitable for manual accessibility validation.

Functional testing focuses on the dynamic behaviour of the document, such as form field responses and script execution. These checks can be automated using simulation frameworks like Selenium, Robot Framework, or PDF parsing tools based on iText or PDFBox.

Example of a functional test in Java to verify the value of a form field:

```
PDDocument doc = PDDocument.load(new File("form.pdf"));
PDAcroForm form = doc.getDocumentCatalog().getAcroForm();
PDField field = form.getField("username");
assert field != null && !field.getValueAsString().isEmpty();
doc.close();
```

In parallel, semantic testing aims to verify the consistency between structure and meaning. For example, a field named “email” should contain a value conforming to a standard email pattern. Such tests can be implemented in Python or Java using validation rules or semantic inference models supported by ontologies.

PDF accessibility is validated according to WCAG 2.1 guidelines and ISO 14289-1 (PDF/UA) standards. The main checks include:

- Presence of semantic tags and logical reading order;
- Definition of alternative text for images;
- Correct use of titles and headings;
- Compatibility with screen readers and assistive technologies.

Tools such as PAC 2021 and Adobe Accessibility Checker generate detailed reports and allow direct correction of identified issues. In academic or institutional contexts, these reports are often included as part of compliance documentation.

Integrating PDF testing within Continuous Integration / Continuous Deployment (CI/CD) pipelines ensures consistent quality across all generated documents.

A typical CI/CD workflow includes:

1. Automatic generation of the PDF via scripts or APIs;
2. PDF/A and PDF/UA validation with VeraPDF and PAC CLI;
3. Functional testing of fields and scripts using PDFUnit;
4. Result storage and report generation in XML or JSON.

This strategy allows for early detection of regressions and compatibility issues, significantly reducing publication and archiving errors.

In research or software experimentation contexts, the testing of interactive PDFs can be the subject of quantitative analysis. Commonly adopted metrics include:

Table 13: Evaluation Metrics for the Quality and Accessibility of Interactive PDFs

Metric	Description
Compliance Rate	Percentage of PDFs that meet reference standards
Average Validation Time	Average duration of the testing process per document
Accessibility Index	Summary measure of assistive readability
Average Errors per Category	Distribution of errors by type (layout, tags, scripts, metadata)

These indicators support the continuous improvement of production and validation systems. Interactive PDF testing is no longer a secondary activity, but a strategic component of document processes. The use of open-source tools and CI/CD integration represents the state of the art for ensuring reliability, accessibility, and regulatory compliance. The future will see the emergence of AI-based semantic testing, capable of evaluating not only the technical structure but also the conceptual coherence of content.

Testing and validation activities thus provide the foundation for a final reflection on the state of the art and on the evolutionary perspectives of interactive PDFs. The final chapter presents a synthesis of achieved results, highlighting open challenges and future directions for research and development in the field of intelligent digital documentation.

3.12 Conclusions

This section has provided a systematic analysis of the evolution, characteristics, and potential of the Portable Document Format (PDF), with particular focus on its interactive capabilities and their implementation through modern technologies and software libraries.

From its origins as a static, print-oriented format, the PDF has progressively evolved into a complex and versatile infrastructure, capable of integrating multimedia content, scripting logic, and structured metadata. This evolution has consolidated its role as a universal document standard, internationally recognised through the ISO 32000 family of specifications, which ensure its stability, compatibility, and transparency over time.

The analysis highlighted how the introduction of interactive components, such as fillable forms, annotations, JavaScript, and multimedia resources, has redefined the concept of a digital document, transforming it from a passive object into an active interface for communication and data collection. In parallel, the formalisation of specialised profiles such as PDF/A, PDF/X, PDF/E, and PDF/UA has allowed the format to adapt to domain-specific contexts, ensuring quality, accessibility, and interoperability.

From a technical standpoint, the use of libraries such as Apache PDFBox, iText, and OpenPDF demonstrates that it is now possible to automate the entire document lifecycle, from generation to validation, from integration with external systems to compliance checking using tools such as VeraPDF or PAC 2021. These technologies have transformed the PDF into a convergence node between content, data, and digital services.

Looking ahead, the evolution of interactivity in PDFs will inevitably be influenced by the integration of artificial intelligence (AI) and machine learning (ML) technologies. Already today, semantic analysis algorithms and neural networks are used for automatic knowledge extraction, content classification, and structured metadata generation. The most promising directions include:

- The use of Large Language Models (LLMs) to interpret and dynamically respond to the content of interactive documents;
- The automatic personalisation of PDFs based on user profiles or preferences;
- The semantic enrichment of files through integration with ontologies and knowledge graphs;
- The predictive analysis of document usage patterns to improve usability and accessibility;
- The development of intelligent assistants integrated into PDF viewers, capable of guiding users in form completion or content understanding.

This convergence between PDF and artificial intelligence marks the beginning of a new phase in which the document is no longer merely a container of information, but a cognitive unit capable of interacting, learning, and adapting. In this scenario, the main challenge will be to balance innovation and standardisation, ensuring that intelligent extensions remain compatible with ISO specifications and respect principles of security and accessibility.

The interactive PDF today represents both a mature and consolidated platform and a rapidly evolving field. Artificial intelligence technologies and the growing attention to digital sustainability open the way to new paradigms of intelligent documentation, where the PDF format can act as an interface between humans and cognitive systems.

From this perspective, the document of the future will no longer be merely a medium for information transmission, but an active participant in the knowledge cycle, capable of engaging in interpretation, decision-making, and collective learning.

References

- Adobe Systems Incorporated. (2024). Acrobat JavaScript Scripting Guide. Adobe Inc. https://www.adobe.com/devnet/acrobat/javascript.html
- Adobe Systems Incorporated. (2023). PDF Reference, Sixth Edition, Version 1.7. Adobe Inc. https://opensource.adobe.com/dc-acrobat-sdk-docs/
- Apache Software Foundation. (2025). Apache PDFBox Documentation. https://pdfbox.apache.org
- European Commission. (2023). EN 301 549 – Accessibility Requirements for ICT Products and Services. Publications Office of the European Union. https://ec.europa.eu
- International Organization for Standardization. (2008). ISO 32000-1:2008 – Document management – Portable document format – Part 1: PDF 1.7. ISO. https://www.iso.org/standard/51502.html
- International Organization for Standardization. (2020). ISO 32000-2:2020 – Document management – Portable document format – Part 2: PDF 2.0. ISO. https://www.iso.org/standard/75839.html
- International Organization for Standardization. (2012). ISO 19005-1:2012 – Document management – Electronic document file format for long-term preservation – PDF/A-1. ISO.
- International Organization for Standardization. (2014). ISO 14289-1:2014 – Document management applications – Electronic document file format enhancement for accessibility – PDF/UA-1. ISO.
- International Organization for Standardization. (2017). ISO 15930-8:2017 – Graphic technology – Prepress digital data exchange using PDF – PDF/X-8. ISO.
- International Organization for Standardization. (2008). ISO 24517-1:2008 – Document management – Engineering document format using PDF – PDF/E-1. ISO.
- Open Preservation Foundation. (2024). VeraPDF User Guide and Technical Reference. https://verapdf.org/documentation/
- PDF Association. (2024). PDF 2.0 Reference and Implementation Notes. PDF Association. https://www.pdfa.org
- PDF Association. (2023). Best Practices for Accessible PDF (PDF/UA). PDF Association. https://www.pdfa.org/resource/pdfua-best-practices
- PDF Tools AG. (2024). PAC 2021 – PDF Accessibility Checker Documentation. https://www.access-for-all.ch/en/pdf-accessibility-checker-pac.html
- Schlosser, M., & Hellwig, F. (2023). Accessible and Interactive PDF Documents in Digital Workflows. Journal of Digital Publishing Technologies, 19(3), 45–58.
- Tzovaras, D., & Economou, D. (2022). Intelligent Document Processing: AI Integration in PDF Workflows. International Journal of Digital Transformation, 11(2), 75–92.

4. Interactive reporting and visualizations

Irritable bowel syndrome (IBS) is a common functional gastrointestinal disorder characterized by recurrent abdominal pain and changes in bowel habits. The prevalence of the syndrome worldwide is estimated at between 10% and 15% of the population, with European figures ranging from 7 to 12% of adults [24]. The incidence is higher in women (2:1) and onset is more frequent between the ages of 20 and 49, although it can occur at any age [25]. The main symptoms include abdominal pain, bloating, flatulence, and changes in stool consistency and frequency. These are often associated with upper digestive disorders (nausea, slow digestion) and a significant impact on quality of life, with psychosocial and economic repercussions [24].

The gut microbiota is the collection of microorganisms that colonize the human gastrointestinal tract. It is estimated to comprise trillions of microbial cells, with a number of genes (microbiome) approximately 150 times greater than that of the human genome [26]. It is composed mainly of bacteria, but also includes archaea, viruses, and fungi, with a predominance of the phyla Firmicutes, Bacteroidetes, Actinobacteria, and Proteobacteria.

It is shown in the literature that patients with IBS frequently suffer from dysbiosis, characterized by reduced microbial diversity, alterations in the ratio between dominant phyla, and specific variations in genera and species [26-27].

Among the most recurrent observations is the reduction of butyrate-producing bacteria, such as Faecalibacterium prausnitzii, Roseburia spp., and Eubacterium rectale, with a consequent decrease in short-chain fatty acids (SCFA) considered protective [28]. This is often associated with a decrease in beneficial species such as Bifidobacterium and Lactobacillus, whose deficiency is related to a worsening of intestinal symptoms [29].

The recognition of microbiome alterations has led to the development of specific reports, such as those used in this study (e.g., Mikrobiom Maxi NEU analysis) [35]. These reports include parameters such as biodiversity, dysbiosis index, phyla ratio (Firmicutes/Bacteroidetes, Actinobacteria/ Proteobacteria), and concentrations of key bacterial species, allowing for a detailed clinical assessment. By interpreting these data, it is possible to better understand the origin of IBS, and explore personalized therapeutic strategies, including diet (low-FODMAP), probiotics, prebiotics, and, in selected cases, fecal microbiota transplantation [44-46]. Such data are still not easily accessible to non-specialist physicians and patients, who have to face data that is difficult to understand and interpret [30, 39, 43].

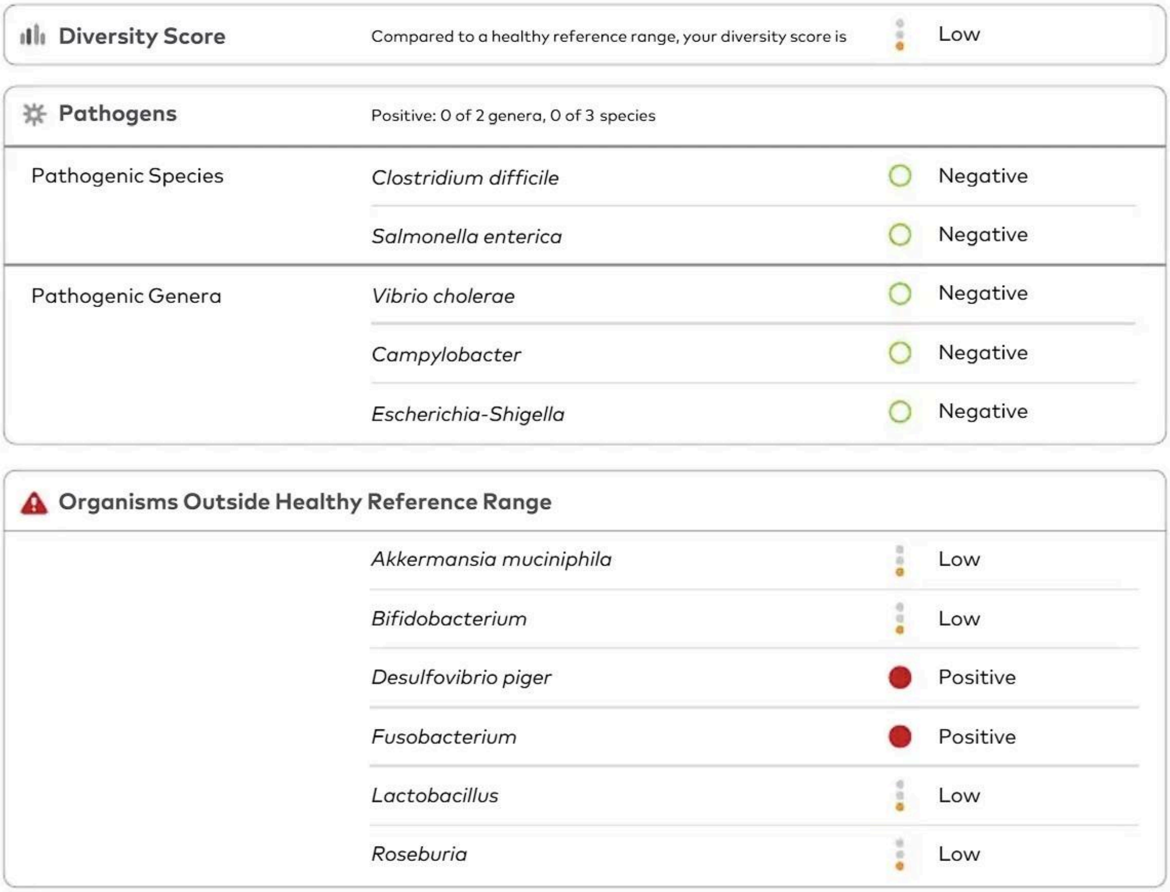
Microbiome testing is an increasingly used clinical tool for describing the composition and functional characteristics of the gut microbiota. Starting from fecal samples analyzed using sequencing methods (mainly 16S rRNA), reports collecting quantitative and qualitative data on the patient's microbial ecosystem are produced. However, diagnosis is complex because it is based on clinical criteria rather than unambiguous markers [37-38]. Imbalances in gut microbiota composition are one of the major causes of IBS, making microbiome analyses increasingly important tools [47]. The microbiota composition can be retrieved by analyzing stool samples of a patient. The report of the analysis contains information about the sample, the microbiota composition, the analysis results and can also contain suggestions about what to do and suggestions about the life style of the patient to mitigate or prevent issues. However, these reports are rich of technical data that are not always easy to interpret as they require domain knowledge and lack user-friendly visual structures that support clinical interpretation [30], [39]. Phan et al. highlighted the importance of having clear and easily interpretable diagnostics [27]. This paper proposes several indications about what to show to the different users and how to do it, including some interactive features. We started from the analysis of existing reports, inquiring experts in the field of diagnostic labs to highlight the main characteristics a dynamic report should have.

4.1 Existing works

A microbiome report presents different levels of information, allowing for a multi-level assessment of the microbiota. The properties of the sample, such as color, consistency, and pH of the stool are macroscopic parameters that provide a preliminary description of the state of the sample and may suggest possible functional alterations of the intestinal tract. Synthetic indices, consisting of statistical measures such as biodiversity indices (e.g., Shannon, Simpson), the dysbiosis index, and phylum ratios, such as the Firmicutes/Bacteroidetes ratio, are global indicators of microbiome balance or imbalance and provide an immediate overview of the patient's condition. The distribution of bacterial populations at the phylum, genus, and species levels reveals both microorganisms considered protective, such as Faecalibacterium prausnitzii and Bifidobacterium, and those potentially pathogenic, such as Klebsiella or other Enterobacteriaceae. Metabolic parameters are indirect markers of the functional activities of the microbiota. These include the production of short-chain fatty acids, bile acids, indoles, and biogenic amines, all of which contribute to the metabolic profile of the microbial community. Such information allows the analyst to move from simple, macroscopic observations to high-resolution molecular and quantitative data, providing an overall picture of the state of the gut microbiota.

Despite the wealth of information they provide, traditional microbiome reports have several limitations. One of the main issues is that they are static, so data filtering or display customization based on the clinical context and the user's needs is not possible. The amount of numerical and taxonomic data can be complex to interpret for patients and non-specialist physicians: indices, ratios, and names of microorganisms often require advanced knowledge of microbiology. The tabular and graphical structure is detailed, but does not help to quickly understand the meaning of the reported values. Moreover, static reports lack comparative tools (e.g., comparisons with previous reports for the same patient, or comparisons with a reference cohort) and do not allow for the dynamic highlighting of correlations between parameters. These limitations motivated the need to develop dynamic, more accessible, and interactive interfaces.

uBiome offered gut microbiota analysis directly to consumers, using a kit for collecting fecal samples and an online portal. The company provided reports showing the composition of the microbiota and offered comparisons with their reference databases [48]. Figure 1 shows an example of a report highlighting the bacterial diversity index and the presence of potential pathogens. The section shows the values calculated for the user and a comparison with a reference range considered “healthy”. In the example of Figure 1, the diversity score is compared to a healthy reference range, and it shows that the score is low (only three values are presented). As visible in the figure, the report has a predefined structure that does not speed up the reading of the bad results consultation.



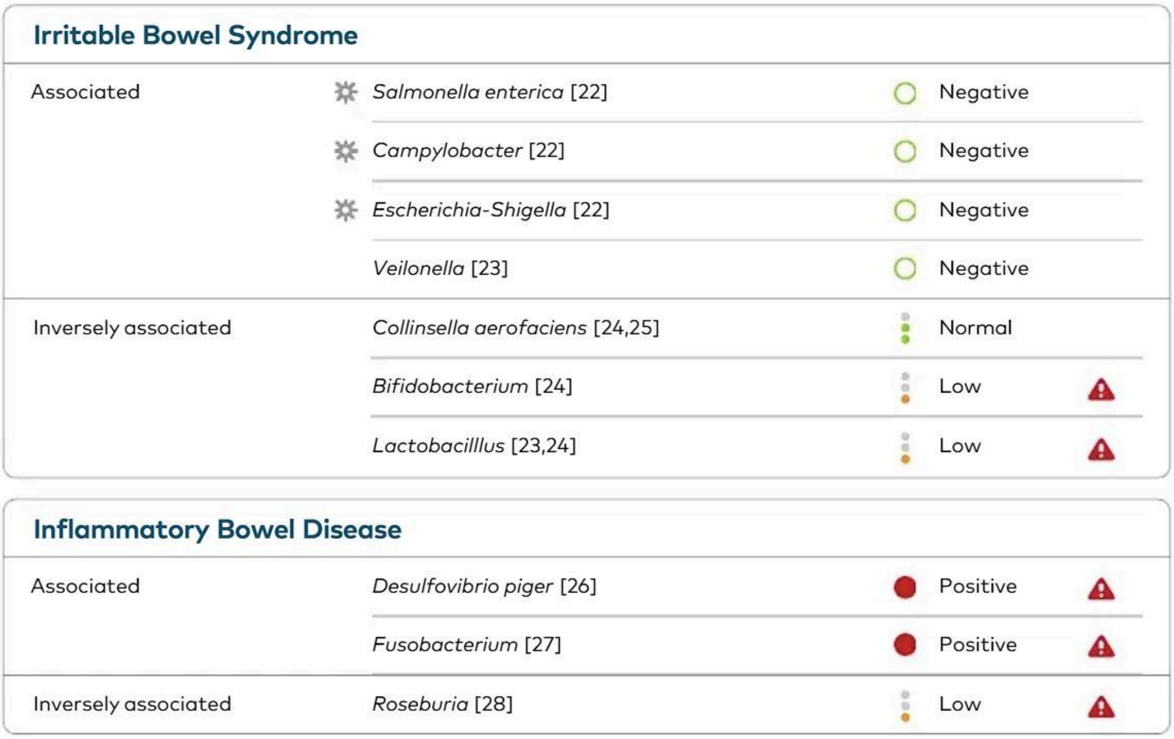
Diversity Score	Compared to a healthy reference range, your diversity score is	⋮	Low

Pathogens	Positive: 0 of 2 genera, 0 of 3 species		
Pathogenic Species	*Clostridium difficile*	○	Negative
	Salmonella enterica	○	Negative
Pathogenic Genera	*Vibrio cholerae*	○	Negative
	Campylobacter	○	Negative
	Escherichia-Shigella	○	Negative

Organisms Outside Healthy Reference Range		
Akkermansia muciniphila	⋮	Low
Bifidobacterium	⋮	Low
Desulfovibrio piger	●	Positive
Fusobacterium	●	Positive
Lactobacillus	⋮	Low
Roseburia	⋮	Low

Fig. 1: Example of a uBiome report showing bacterial diversity index, detection of potentially pathogenic species, and organisms outside the reference ranges.

In Figure 1, only in the bottom half are bad parameters and Figure 2 shows another part of the report that links specific microorganisms to known diseases or clinical conditions, such as irritable bowel syndrome (IBS) or inflammatory bowel disease (IBD). In this section of the report, the data are organized into “associated” and “inversely associated,” with a coherent visual alert system (colors and symbols) to highlight deviations from normal values. Also in this case, the structure is predefined and good data (e.g., negative values) are presented before bad ones, which are those the user is most interested in, at least at first sight.



Irritable Bowel Syndrome

Associated	✱ *Salmonella enterica* [22]	○	Negative	
	✱ *Campylobacter* [22]	○	Negative	
	✱ *Escherichia-Shigella* [22]	○	Negative	
	Veilonella [23]	○	Negative	
Inversely associated	*Collinsella aerofaciens* [24,25]	●	Normal	
	Bifidobacterium [24]	●	Low	⚠
	Lactobacilllus [23,24]	●	Low	⚠

Inflammatory Bowel Disease

Associated	*Desulfovibrio piger* [26]	●	Positive	⚠
	Fusobacterium [27]	●	Positive	⚠
Inversely associated	*Roseburia* [28]	●	Low	⚠

Fig. 2: Example of a *uBiome* report showing microorganisms associated with IBS and IBD, divided into positive, negative, and inversely associated.

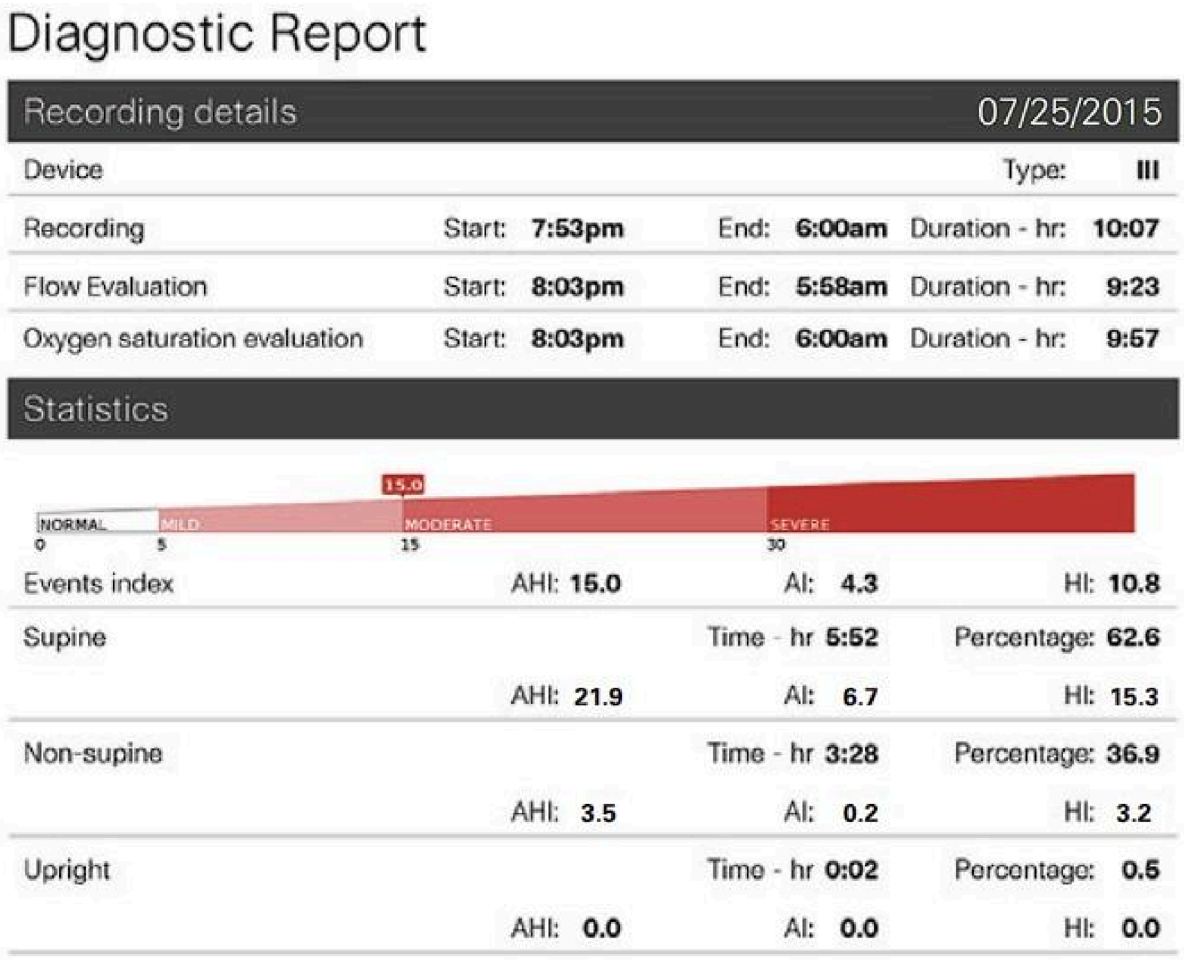
uBiome's approach aimed to simplify the communication of results, favoring an accessible and intuitive format for the end user rather than a technical document intended for clinicians.

Viome microbiome tests differ from traditional approaches based on 16S rRNA, as they use metatranscriptomics techniques[1] (RNA sequencing) to analyze not only the composition of the microbiota but also its functional activity [49].

AirView relates to patients undergoing respiratory therapies, such as CPAP (Continuous Positive Airway Pressure)[2] and non-invasive ventilation. It allows clinicians to remotely monitor usage parameters, generate compliance reports, evaluate respiratory events, and view time trends, with the option to export data in PDF format [19]. Even if it is not strictly related to gut microbiota, the interest in this report is related to some features that are interesting for our prototype and are introduced in this report. One feature is the use of text with visualization. When visuals are shown on the screen, it is mandatory to support images with descriptive text that clarifies what is visualized, explains why it is important, and guides the user to the reading of the content. Figure 3 shows an example of a standard diagnostic report. In addition to the recording data (monitoring duration, start and end times), key indices such as the Apnea-Hypopnea Index (AHI), the apnea index (AI), and the hypopnea index (HI) are summarized. These values are stratified according to the patient's position (supine, non-supine, upright), allowing for a targeted assessment of the conditions that favor respiratory events.

[1] The term metatranscriptomics refers to the analysis of all messenger RNAs (mRNAs) expressed by microorganisms present in a sample. Unlike metagenomics, which provides a snapshot of the overall genetic composition, metatranscriptomics allows the analyst to assess which genes are actively expressed and which metabolic functions are occurring at the time of sample collection.

[2] CPAP is a device that delivers a constant flow of positive pressure air through a nasal or nose-mouth mask. This mechanism prevents the upper airway from collapsing during sleep and is the standard treatment for obstructive sleep apnea (OSA).



Diagnostic Report

Recording details						07/25/2015
Device					Type:	III
Recording	Start:	7:53pm	End:	6:00am	Duration - hr:	10:07
Flow Evaluation	Start:	8:03pm	End:	5:58am	Duration - hr:	9:23
Oxygen saturation evaluation	Start:	8:03pm	End:	6:00am	Duration - hr:	9:57

Statistics

Events index	AHI:	15.0	AI:	4.3	HI:	10.8
Supine			Time - hr	5:52	Percentage:	62.6
	AHI:	21.9	AI:	6.7	HI:	15.3
Non-supine			Time - hr	3:28	Percentage:	36.9
	AHI:	3.5	AI:	0.2	HI:	3.2
Upright			Time - hr	0:02	Percentage:	0.5
	AHI:	0.0	AI:	0.0	HI:	0.0

Fig. 3: Example of a diagnostic report generated by AirView: recording details, overall AHI and its distribution by position are reported, together with apnea and hypopnea indices.

Figure 4 shows another part of the report that adds graphical representations of respiratory events during the night. In addition to body position, the graph tracks the temporal distribution of different types of apnea (obstructive, central, mixed), hypopneas, desaturations, and snoring. This visualization allows the reader to immediately grasp the trend of events and their severity, integrating the numerical values of the standard report [36].

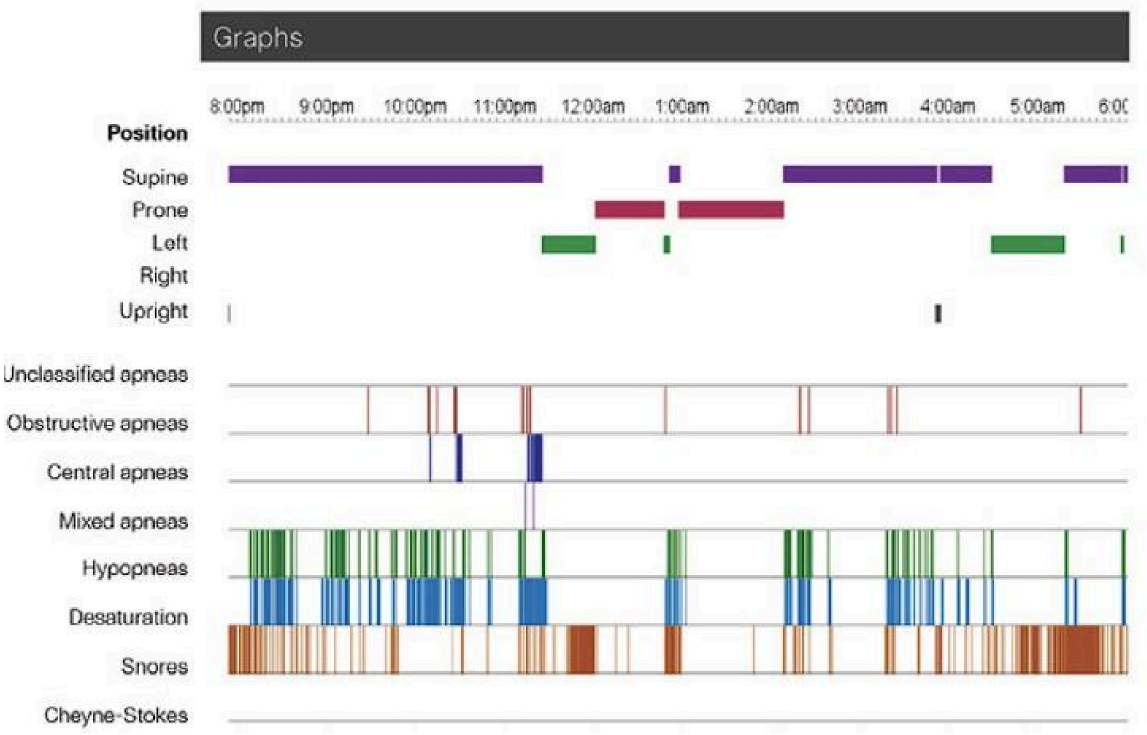


Fig. 4: Example of a detailed AirView report: temporal distribution of apneas (obstructive, central, mixed), hypopneas, desaturation episodes, and snoring, stratified according to body position.

Several issues emerged from the analysis of existing reports and the interview with domain experts. The reports are not easily reproducible due to differences in sampling protocols, laboratory techniques, bioinformatic approaches, and statistical tools used. Some interesting characteristics have been considered during the analysis of the existing reports and from the interaction with the end users. One explicit need emerged, which has not been implemented in the analyzed reporst is that the patient, as well as the physicians, want to read anomalies immediately, then see the other parameters. From this side, the uBiome report showed some rigidity in the presentation order of the results. The interpretation of data and associations between microbes and clinical conditions is another aspect. In the analyzed report and during the interviews, it emerged that people want to correlate data, so they want to put together data that are relevant to stay together. Reports show correlations between microbial taxa and symptoms, but these do not necessarily imply a causal relationship. Often, these relationships are derived from descriptive observations rather than controlled clinical studies, which can generate misleading expectations or interpretations [26].

Another issue, which is not relevant for the report but is relevant for the data results, is the reliance on reference databases. When in the report a reader sees "healthy ranges", it is not clear what the ranges are and which individuals or samples are correlated. The ranges are often constructed on cohorts with demographic, dietary, or geographic characteristics that may differ from those of the subject being analyzed, introducing potential biases and reducing the clinical validity of the comparison [32].

Also, the complexity of the language and graphic representation can be an issue. Reports are often written with a high degree of technicality—statistical indices, specialist terms, and graphs without explanations—making them difficult to understand for non-experts, including general practitioners and non-microbiologist specialists. The documents produced by laboratories often contain technical terms, statistical indices, and graphical representations that are difficult to understand without adequate explanation. Indices such as Shannon biodiversity, the Firmicutes/Bacteroidetes ratio, or the dysbiosis index are presented as numerical values, but their clinical significance or how they should be interpreted is rarely clarified [30], [40]. This generates the need to provide information with different styles and skill levels. According to the user who is reading the report, the language should change. This is reachable both by using a dynamic report and a static one, provided that in the static report (e.g., the printed version) there are clues that let the reader jump to the part of the report where the explanation (in the language that the user understands) is provided. For some indices, it is not easy to understand how they are created. While the patient might not be aware and not interested in how such data are generated, the physicians, the laboratory experts, and the experts in general require this information because it allows them to better evaluate the value that appears on the report and better correlate with other values. Also, the external information, such as dietary habits or use of drugs, can affect the diagnosis.

In addition to these technical aspects, there are also ethical and privacy issues: microbiome data is extremely sensitive, and reports suggesting possible diagnoses without adequate validation can have legal or psychological implications for the user. This difficulty is clearly reflected in some of the analyzed examples. The reports from uBiome, for example, showed colorful graphs and associations between specific microorganisms and diseases such as IBS or IBD, without adequately distinguishing between correlation and causation [23]. In other cases, such as Viome, the information was synthesized into health “scores”, with the aim of making it easier for patients to understand, but without sufficient transparency on the calculation criteria: the result is a report that is perceived as too simplistic for clinicians and, at the same time, unreliable for end users. Even research-oriented services, despite offering extremely detailed data of high scientific quality, are not designed for direct use by patients or non-specialist physicians: the sheer volume of information and the absence of explanatory notes limit their immediacy.

As stated before, making comparisons with reference cohorts has been revealed as a mandatory feature because, in this way, it is easier to position the sample characteristics with similar profiles. Most reports provide data relating to individual subjects, without integrating tools that allow the results to be placed within a broader population, divided by age, gender, clinical status, or geographical area. This aspect is relevant in giving clinical meaning to the data; for example, a given biodiversity index value can have very different implications when compared with healthy cohorts, patients with irritable bowel syndrome, or other gastrointestinal diseases [27], [41].

The analyzed examples show partial approaches to this problem. uBiome allowed comparison with a generic reference database, but without the possibility of selecting specific cohorts or checking the methodological quality of the sample [31]. Viome offers summary scores, but the reference base is not always clearly explained, and the average values cannot be stratified by clinical subgroups. In Microbiome Insights, the data is produced with scientific rigor and can be compared at the research level, but the reports are not designed to offer a direct and immediate comparison with specific clinical cohorts.In other medical fields, there are platforms that systematically integrate comparisons with reference populations: one example is ResMed AirView, which allows clinicians to evaluate respiratory parameters by comparing them with normative ranges and aggregated usage data [33]. The absence of similar functionalities in the microbiome field represents an important difference, which limits clinical interpretation and the possibility of contextualizing individual results.

Analysis of the issues highlighted in existing reports highlights the need for a paradigm shift: from the current static model, tied to poorly interactive PDF documents, to digital tools capable of integrating scientific complexity and usability. The evolution towards dynamic reports is part of the current digital transformation in medicine, where the adoption of interactive platforms is now an established direction [42].

In this scenario, microbiome reports have the potential to become not only a static support for diagnosis but also actual interactive analysis tools, capable of adapting to the clinical context and user level (physician, researcher, patient). The goal is to transform data into usable knowledge, creating a clearer, more timely, and personalized flow of information.

Despite technological advances and growing interest in the microbiota, microbiome reports still lag significantly behind other clinical areas. On the one hand, platforms such as ResMed AirView have already introduced remote monitoring systems and interactive reports; on the other hand, static solutions remain prevalent in microbiomics, geared more towards research than everyday clinical practice.

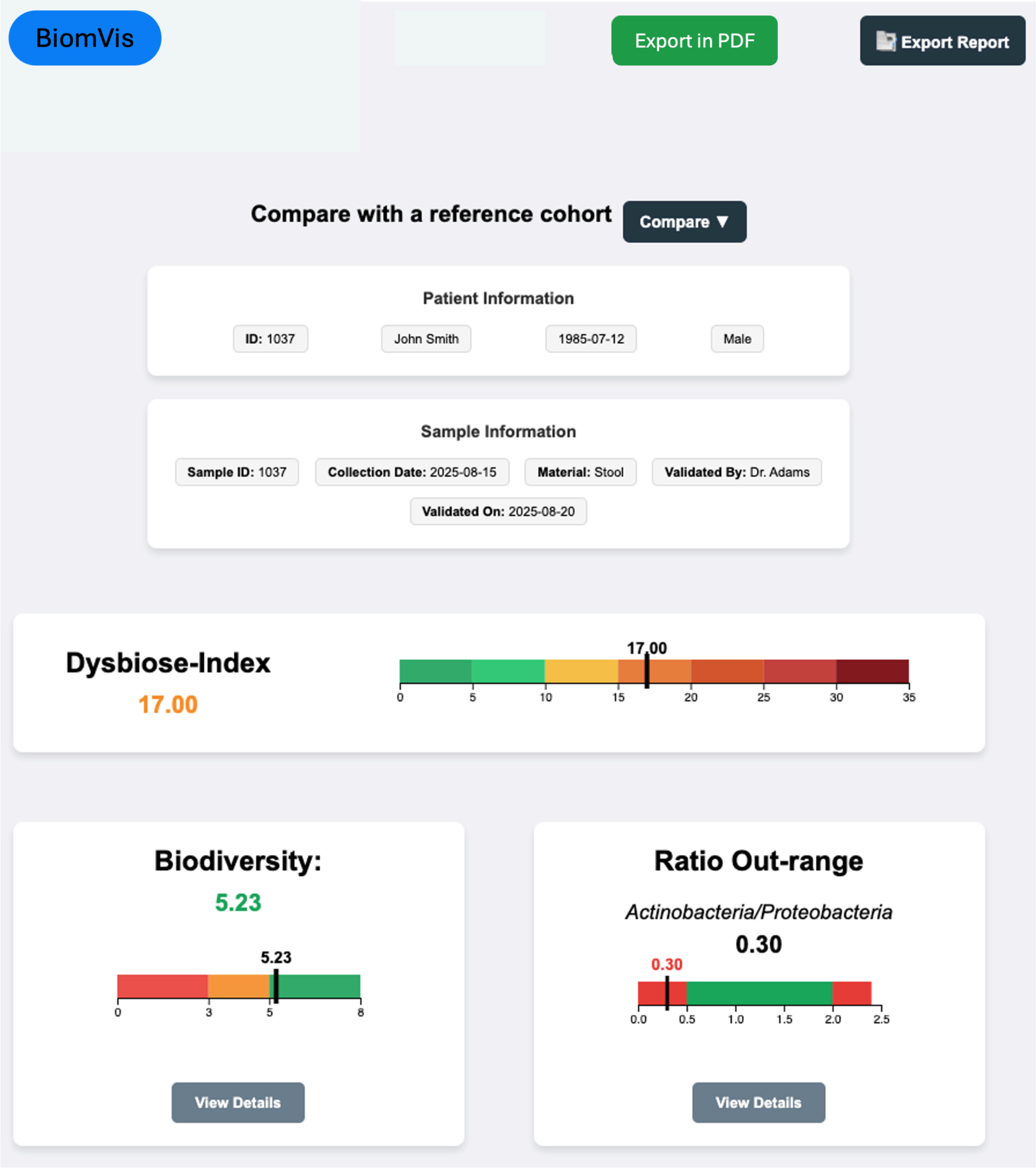
The discussed examples confirm that a reporting tool should be disseminated with a solid methodological basis, provide intuitive scores, with transparency and reproducibility, and provide accurate data that is also readable by non-specialists.

4.2 A report prototype

The prototype interface has been designed to follow a sequential and intuitive organization, accompanying the user from an overview of the report to the most specific details.

The initial section is dedicated to basic information (patient data and sample characteristics), followed by thematic blocks dedicated to summary indices, macroscopic properties, taxonomic distribution, and metabolites. Each section is presented in the form of cards or independent graphic modules, making consultation immediate and facilitating comparison between parameters. An example is shown in Figure 5.

Associated to each card there is a button that allows the user to know more about the index shown in the card. This is a feature that can help people who need to understand the indices but don’t have enough knowledge.



Compare with a reference cohort

Patient Information

ID: 1037 | John Smith | 1985-07-12 | Male

Sample Information

Sample ID: 1037 | Collection Date: 2025-08-15 | Material: Stool | Validated By: Dr. Adams | Validated On: 2025-08-20

Dysbiose-Index 17.00

Biodiversity: 5.23

Ratio Out-range

Actinobacteria/Proteobacteria

0.30

Fig. 5: System prototype

The visualization of multiple parameters can be confusing if there are many good data points and some bad ones. The implementation logic of the report is that the reader must immediately see the out-of-range parameters, then, by clicking on the relative button, the interface shows all parameters. Fig. 6 shows two different moments during the interaction with the prototype: on the left the initial state, where only problematic parameters are visible; on the right, the expanded selection of parameters after clicking on the “Include in-range-data” of Phyla group of parameters.

The ranges, when possible, must have the same interval so that the data are more comparable, and this makes it clear which are the good and bad interval values. In the case of very small ranges, by pressing on the button “Rescaling” the the scale of each parameter changes to extend the in-range area and make more visible the position of the results in that range.

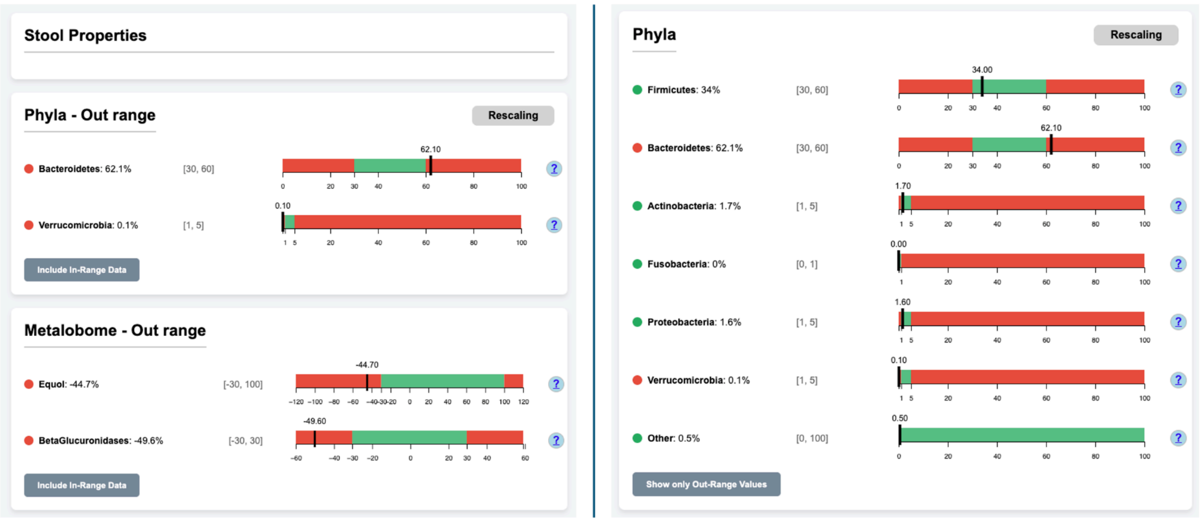
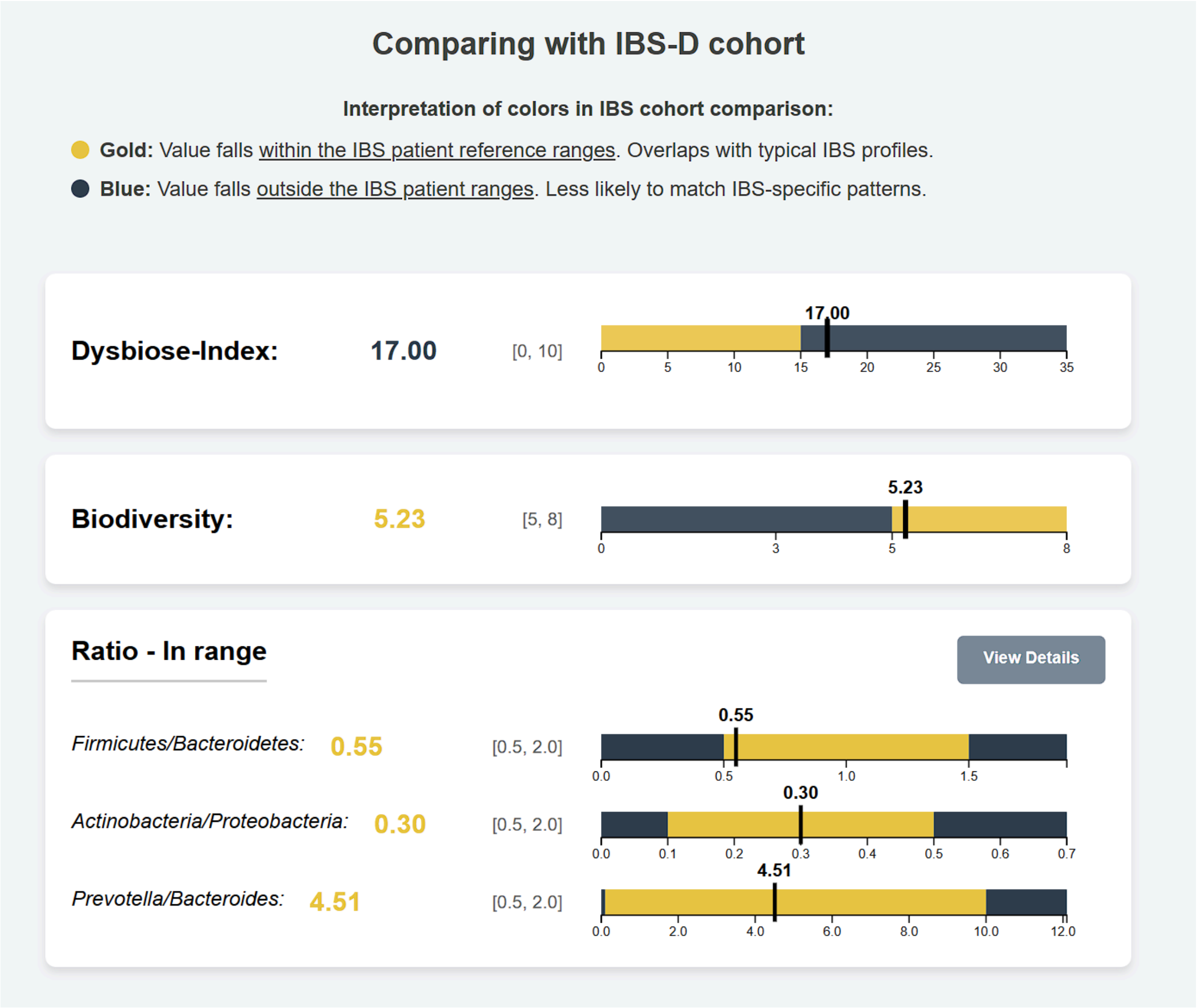


Fig. 6: Compressed and detailed visualization of stool properties

In order to let the patient understand how a parameter behaves according to a reference range, a “Compare” button (see Fig. 5, next to “Compare with a reference cohort) transforms the view into a typical range of the cohort and the value is positioned into the domain of the specific parameter, revealing how that patient relates to the reference range of patients.



Comparing with IBS-D cohort

Interpretation of colors in IBS cohort comparison:

- ● **Gold:** Value falls within the IBS patient reference ranges. Overlaps with typical IBS profiles.
- ● **Blue:** Value falls outside the IBS patient ranges. Less likely to match IBS-specific patterns.

Dysbiose-Index:	17.00	[0, 10]
Biodiversity:	5.23	[5, 8]

Ratio - In range

View Details

Firmicutes/Bacteroidetes:	0.55	[0.5, 2.0]
Actinobacteria/Proteobacteria:	0.30	[0.5, 2.0]
Prevotella/Bacteroides:	4.51	[0.5, 2.0]

Fig. 7: Example of patient's parameters comparison with a reference cohort

Fig. 6 shows the visualization of the comparison of the patient with a reference cohort. The colors are different to help the user understanding that the reported intervals are not those of the ideal range but the values of the cohort of patients similar to the reference patient.

4.3 Assessment of the first version

To verify the effectiveness of the prototype developed, an informal evaluation was conducted involving a small group (n=8) of users, consisting of simulated patients and nonspecialist physicians. The aim was to observe how performance changes when using the traditional report compared to the proposed prototype. Particular attention was paid to the speed with which users were able to identify abnormal parameters, the number of errors made, and the perceived cognitive load.

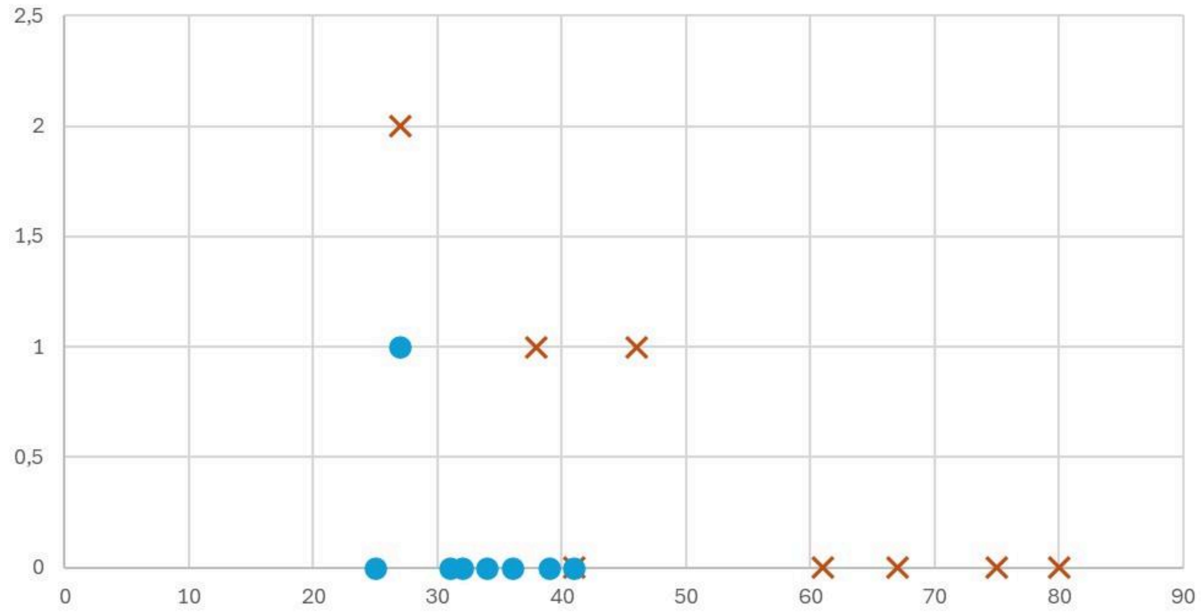


Fig. 7: Errors and time spent to read the traditional report vs the proposed prototype. Blue represents the paper-based report, while the red cross represents the prototype. The X-axis shows the time to complete the task (in seconds), and the Y-axis shows the number of errors.

Errors and times in traditional reports. Figure 7 shows the result of the test performed with the involvement of the users on the traditional report, represented with red crosses; the relationship between the time taken to identify abnormal parameters and the errors made in reading traditional reports are visible. The times are quite variable, with some users taking more than a minute to complete the task. In several cases, greater speed was associated with a higher number of errors, a sign that reading the report required considerable interpretative effort. Conversely, participants who spent more time on the analysis tended to make fewer errors, suggesting that more careful reading compensates for the poor visual clarity of the static format. Overall, this trend indicates that the traditional interface does not always ensure an effective balance between speed and accuracy.

Errors and times in the prototype: Figure 7 shows the analysis carried out on the proposed prototype, represented with blue dots. The performance of the prototype show an improvement: completion times have been significantly reduced and have become more consistent, with most users concentrated in a narrow range between 25 and 40 seconds. Errors are almost entirely absent, a sign that the interactive representation allows for more intuitive and immediate reading, reducing the ambiguities that characterized the static report.

Perceived cognitive load: Figure 8 shows a comparison of the cognitive load, particularly the overall effort, perceived by participants. The traditional report is evaluated as more burdensome, with values that in several cases reach high scores. The prototype, on the other hand, was judged to be easier to interpret: most users reported reduced effort, often two or three points lower than the static report. This difference is particularly significant because it suggests that the interface reduces time and errors and makes the reading experience smoother and less tiring.

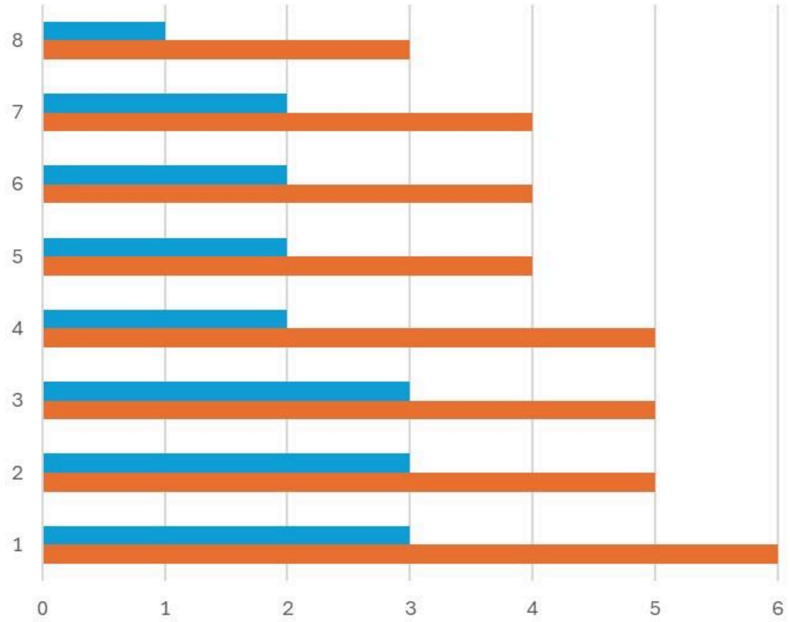


Fig. 8: Comparison of perceived cognitive load (x-axis), for each participant (y-axis) between traditional report (orange) and prototype (blue).

The prototype allows for faster, more accurate, and less laborious data consultation than traditional reports, confirming the validity of the adopted design solutions.

Conclusions and Next Steps

This deliverable reports the status of the activities at month 15th. The activities are proceeding according to the plans, and specifically, a dynamic report showing patient's data and comparing more patients at once have been developed. The dynamic report, allow the user to make selections and filters in order to rapidly get to the desired information. The report presents the information in three different ways: 1) dynamic report through a web browser; 2) dynamic report as a PDF; 3) printable report with cross references in the text. In all cases, the user can quickly get to the desired information.

Next steps will cover the analytical process. For instance, a result of the stool analysis is a .fastq file, which contains the required information. The processing of a .fastq file requires dedicated tools and also dedicated skills. The aim for the future is to integrate stool sample analysis and information to provide better analysis results.

References

[1] Krause, T.; Jolkver, E.; Bruchhaus, S.; Kramer, M.; Hemmje, M. GenDAI—AI-Assisted Laboratory Diagnostics for Genomic Applications. In Proceedings of the 2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Houston, TX, USA, 9–12 December 2021

[2] Marco X. Bornschlegl, Kevin Berwind, Michael Kaufmann, Felix C. Engel, Paul Walsh, Matthias L. Hemmje, and Ruben Riestra. "IVIS4BigData: A Reference Model for Advanced Visual

Interfaces Supporting Big Data Analysis in Virtual Research Environments”. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 10084 LNCS (2016), pages 1–18. issn: 16113349. doi: 10.1007/978-3-319-50070-6_1.

[3] Thoralf Reis, Marco X. Bornschlegl, and Matthias L. Hemmje. "AI2VIS4BigData: Qualitative Evaluation of an AI-Based Big Data Analysis and Visualization Reference Model". In: Advanced Visual In-terfaces. Supporting Artificial Intelligence and Big Data Applications. Volume 12585. Lecture Notes in Computer Science. Springer International Publishing, 2021, pages 136-162. ISBN: 9978-3-030-68006-0. DOI: 10.1007/978-3-030-68007-7_9.

[4] Leader Studio, UC Berkeley, “ubiome case study.” https://leaderstudio. berkeley.edu/wp-content/uploads/2021/01/ uBiome-Case-Study FINAL. pdf, 2021. Accessed: 2025-09-16.

[5] J. Tap, M. Derrien, H. Tornblom, R. Brazeilles, S. Cools-Portier, J. Dorë,´ S. Storsrud, B. Le Nev¨ e, L. Ohman, and M. Simr´ en, “Identification of an´ intestinal microbiota signature associated with severity of irritable bowel syndrome,” Gastroenterology, vol. 152, no. 1, pp. 111–123, 2017.

[6] A. W. K. Yeung et al., “The promise of digital healthcare technologies.,” Frontiers in public health, vol. 11, p. 1196596, 2023.

[7] O’Donoghue, S.I.; Gavin, A.C.; Gehlenborg, N.; Goodsell, D.S.; Hériché, J.K.; Nielsen, C.B.; North, C.; Olson, A.J.; Procter, J.B.; Shattuck, D.W.; et al. Visualizing biological data-now and in the future. Nat. Methods 2010, 7, S2–S4.

[8] Cruz, A.; Arrais, J.P.; Machado, P. Interactive and coordinated visualization approaches for biological data analysis. Briefings Bioinform. 2019, 20, 1513–1523

[9] Kerren, A.; Schreiber, F. Network Visualization for Integrative Bioinformatics. In Approaches in Integrative Bioinformatics; Chen, M.,Hofestädt, R., Eds.; Springer: Berlin/Heidelberg, Germnay, 2014; pp. 173–202.

[10] Kanehisa, M.; Furumichi, M.; Sato, Y.; Kawashima, M.; Ishiguro-Watanabe, M. KEGG for taxonomy-based analysis of pathways and genomes. Nucleic Acids Res. 2023, 51, D587–D592.

[11] Ropinski, T.; Oeltze, S.; Preim, B. Survey of glyph-based visualization techniques for spatial multivariate medical data. Comput. Graph. 2011, 35, 392–401.

[12] Suschnigg, J.; Mutlu, B.; Koutroulis, G.; Sabol, V.; Thalmann, S.; Schreck, T. Visual Exploration of Anomalies in Cyclic Time Series Data with Matrix and Glyph Representations. Big Data Res. 2021, 26, 100251.

[13] Kammer, D.; Keck, M.; Grunder, T.; Maasch, A.; Thom, T.; Kleinsteuber, M.; Groh, R. Glyphboard: Visual Exploration of High-Dimensional Data Combining Glyphs with Dimensionality Reduction. IEEE Trans. Vis. Comput. Graph. 2020, 26, 1661–1671.

[14] Keim D. A.; Mansmann F.; Schneidewind J.; Ziegler H. (2006). Challenges in Visual Data Analysis. Tenth International Conference on Information Visualisation (IV'06), pp.9–16. Tenth International Conference on Information Visualisation (IV'06), London, England, 05-07 July 2006. https://doi.org/10.1109/IV.2006.31.

[15] Zgraggen E.; Galakatos A.; Crotty A.; Fekete J.-D.; Kraska T. (2017). How Progressive Visualizations Affect Exploratory Analysis. In: IEEE transactions on visualization and computer graphics 23(8), pp.1977–1987. https://doi.org/10.1109/TVCG.2016.2607714.

[16] Rind A.; Wang T. D.; Aigner W.; Miksch S.; Wongsuphasawat K.; Plaisant C., et al. (2013). Interactive Information Visualization to Explore and Query Electronic Health Records. In: Foundations and Trends in Human–Computer Interaction 5(3), pp.207–298. https://doi.org/10.1561/1100000039.

[17] Wang Q.; Laramee R. S. (2022). EHR STAR: The State-Of-the-Art in Interactive EHR Visualization. In: Computer Graphics Forum 41(1), pp.69–105. https://doi.org/10.1111/cgf.14424.

[18] La Rosa B.; Blasilli G.; Bourqui R.; Auber D.; Santucci G.; Capobianco R., et al. (2023). State of the Art of Visual Analytics for eXplainable Deep Learning. In: Computer Graphics Forum. https://doi.org/10.1111/cgf.14733.

[19] Kandel S.; Heer J.; Plaisant C.; Kennedy J.; van Ham F.; Riche N. H., et al. (2011). Research directions in data wrangling: Visualizations and transformations for usable and credible data. In: Information Visualization 10(4), pp.271–288. https://doi.org/10.1177/1473871611415994.

[20] Ribeiro M. T.; Singh S.; Guestrin C. (2016). "Why Should I Trust You?". In: Krishnapuram, Balaji; Shah, Mohak; Smola, Alex; Aggarwal, Charu; Shen, Dou; Rastogi, Rajeev. KDD2016, pp.1135–1144. KDD '16: The 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco California USA, 13 08 2016 17 08 2016. https://doi.org/10.1145/2939672.2939778.

[21] Selvaraju R. R.; Cogswell M.; Das A.; Vedantam R.; Parikh D.; Batra D. (2020). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In: International Journal of Computer Vision 128(2), pp.336–359. https://doi.org/10.1007/s11263-019-01228-7.

[22] Lundberg S. M.; Lee S.-I. (2017). A Unified Approach to Interpreting Model Predictions. Proceedings of the 31st International Conference on Neural Information Processing Systems, pp.4768–4777.

[23] Cruz (2019). Interactive and coordinated visualisation approaches for biological data analysis. In: Briefings in Bioinformatics 20(4), pp.1513–1523. https://doi.org/10.1093/bib/bby019.

[24] E. M. Quigley, P. Bytzer, R. Jones, and F. Mearin, “Irritable bowel syndrome: The burden and unmet needs in europe,” Digestive and Liver Disease, vol. 38, no. 10, pp. 717–723, 2006.

[25] L. Chang, B. B. Toner, S. Fukudo, E. Guthrie, G. R. Locke, N. J. Norton, and A. D. Sperber, “Gender, Age, Society, Culture, and the Patient’s Perspective in the Functional Gastrointestinal Disorders,” Gastroenterology, vol. 130, no. 5, pp. 1435–1446, Apr. 2006, publisher: Elsevier. [Online]. Available: https://doi.org/10.1053/j.gastro.2005.09.071

[26] J. P. Jacobs, V. Lagishetty, M. C. Hauer, J. S. Labus, T. S. Dong et al., “Multi-omics profiles of the intestinal microbiome in irritable bowel syndrome and its bowel habit subtypes,” Microbiome, vol. 11, no. 5, p. 5, 2023

[27] V. C. Phan et al., “Alterations in gut microbiome composition and function in irritable bowel syndrome and increased relative abundance of firmicutes and actinobacteria,” mSystems, vol. 6, no. 1, pp. e00 962–20, 2021.

[28] Y. Feng and D. Xu, “Short-chain fatty acids are potential goalkeepers of atherosclerosis,” Frontiers in Pharmacology, vol. Volume 14 - 2023, 2023. [Online]. Available: https://www.frontiersin.org/journals/pharmacology/articles/10.3389/fphar.2023.1271001

[29] F. Di Vincenzo, A. Del Gaudio, V. Petito, L. R. Lopetuso, and F. Scaldaferri, “Gut microbiota, intestinal permeability, and systemic inflammation: a narrative review,” Intern Emerg Med, vol. 19, no. 2, pp. 275–293, Jul. 2023

[30] M. Leske, J. Fitzgerald, K. Coughlan, T. Krause, M. Hemmje, F. Bottacini, H. Afli, and B. Andrade, “Genomic language models applied for bacterial metataxonomy,” in Proceedings of the 2025 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBM Workshops). Wuhan, China: IEEE, Dec. 2025, workshop on Genomic Foundation Models for Diagnostic Innovation (GFM4D 2025), co-located with IEEE BIBM 2025, to appear.

[31] C. H. King, H. Desai et al., “Baseline human gut microbiota profile in healthy people and standard reporting template,” PLoS One, vol. 14, no. 9, p. e0206484, Sep. 2019.

[32] Mavis, “Ubiome smartgut sample report,” Scribd, 2018, last access: October 2025. [Online]. Available: https://www.scribd.com/document/ 390317393/UBiome-SmartGut-Sample-Report

[33] ResMed, “Airview report guide — sample diagnostic and therapy reports,” https://resmedwebinars.com/assets/uploads/AirView Report Guide 1018991.pdf, 2014, accessed: 2025-09-15

[34] Y. Bhattarai, D. A. Muniz Pedrogo, and P. C. Kashyap, “Irritable bowel syndrome: a gut microbiota-related disorder?” American Journal of Physiology-Gastrointestinal and Liver Physiology, vol. 312, no. 1, pp. G52–G62, 2017, pMID: 27881403. [Online]. Available: https://doi.org/10.1152/ajpgi.00338.2016

[35] P. Tamla, T. Krause, M. Hemmje, F. Monti, F. De Luzi, M. Mecella, B. Andrade, P. Buono, and A. Molinari, “The gendai cloud-native infrastructure and data stewardship for clinical metagenomic diagnostics,” in Proceedings of the 2025 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBM Workshops). Wuhan, China: IEEE, Dec. 2025, workshop on Genomic Foundation Models for Diagnostic Innovation (GFM4D 2025), co-located with IEEE BIBM 2025, to appear. [Online]. Available: https://sites.google.com/view/gfm4d2025

[36] S. Zhu, S. Liu, H. Li, Z. Zhang, Q. Zhang, L. Chen, Y. Zhao, Y. Chen, J. Gu, L. Min, and S. Zhang, “Identification of gut microbiota and metabolites signature in patients with irritable bowel syndrome,” Frontiers in Cellular and Infection Microbiology, vol. 9, p. 346, 2019

[37] T. Holvoet, M. Joossens, J. F. V´azquez-Castellanos, and et al., “Fecal microbiota transplantation reduces symptoms in some patients with irritable bowel syndrome with predominant abdominal bloating,” Gastroenterology, vol. 160, no. 1, pp. 145–157.e8, 2021.

[38] C. Mirzayi, A. Renson et al., “Reporting guidelines for human microbiome research: the storms checklist,” Nature Medicine, vol. 27, pp. 1885–1892, 2021. [Online]. Available: https://www.nature.com/articles/s41591-021-01552-x

[39] M. El-Salhy, J. G. Hatlebakk, O. H. Gilja, O. Brændeland, and T. Hausken, “Efficacy of faecal microbiota transplantation for patients with irritable bowel syndrome in a randomised, double-blind, placebo-controlled study,” Gut, vol. 69, no. 5, pp. 859–867, 2019.

[40] Viome, “Publications,” https://www.viome.com/publications, 2023, last accessed: 2025-09-15.

[41] P. Buono, M. F. Costabile, and R. Lanzilotti, “A circular visualization of people’s activities in distributed teams,” J. Vis. Lang. Comput., vol. 25, no. 6, p. 903–911, Dec. 2014. [Online]. Available: https://doi.org/10.1016/j.jvlc.2014.10.025

[42] S. Porcari, S. C. Ng, L. Zitvogel, H. Sokol, R. K. Weersma, E. Elinav, A. Gasbarrini, G. Cammarota, H. Tilg, and G. Ianiro, “The microbiome for clinicians,” Cell, vol. 188, no. 11, pp. 2836–2844, May 2025, publisher: Elsevier. [Online]. Available: https://doi.org/10.1016/j.cell.2025.04.016

[43] G. Ianiro, L. H. Eusebi, C. J. Black, A. Gasbarrini, G. Cammarota, and A. C. Ford, “Systematic review with meta-analysis: efficacy of faecal microbiota transplantation for irritable bowel syndrome,” Alimentary Pharmacology & Therapeutics, vol. 50, no. 3, pp. 240–248, 2019

[44] T. Krause, P. Tamla, A. Leoni, F. Monti, F. De Luzi, J. F. Gerald, B. Andrade, H. Afli, M. Mecella, P. Buono, A. Molinari, and M. Hemmje, “From reads to reports: A vision for a gfm-powered genomic diagnostic platform,” in Proceedings of the 2025 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBM Workshops). Wuhan, China: IEEE, Dec. 2025, workshop on Genomic Foundation Models for Diagnostic Innovation (GFM4D 2025), co-located with IEEE BIBM 2025, to appear.

[45] G. Ricci, P. Buono, T. Krause, P. Tamla, M. Hemmje, F. De Luzi, F. Leotta, A. Marrella, F. Monti, M. Mecella, “Responsible use of ai in genomics and ethical implications,” in Proceedings of the 2025 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBM Workshops). Wuhan, China: IEEE, Dec. 2025, workshop on Genomic Foundation Models for Diagnostic Innovation (GFM4D 2025), co-located with IEEE BIBM 2025, to appear.

[46] P. Tamla, T. Krause, F. D. Luzi, J. Rossi, M. Marinacci, A. Sepielli, M. Calamo, and M. Hemmje, “LLM-driven cloud-based architecture design,” in Proceedings of the 2025 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBM Workshops). Wuhan, China: IEEE, Dec. 2025, workshop on Genomic Foundation Models for Diagnostic Innovation (GFM4D 2025), co-located with IEEE BIBM 2025, to appear.

[47] C. Allaband, D. McDonald, Y. V´azquez-Baeza, J. J. Minich, A. Tripathi, D. A. Brenner, and et al., “Microbiome 101: Studying, analyzing, and interpreting gut microbiome data for clinicians,” Clinical Gastroenterology and Hepatology, vol. 17, no. 2, pp. 218–230, 2019.

[48] C. H. King, H. Desai et al., “Baseline human gut microbiota profile in healthy people and standard reporting template,” PLoS One, vol. 14, no. 9, p. e0206484, Sep. 2019.

[49] Y. Bhattarai, D. A. Muniz Pedrogo, and P. C. Kashyap, “Irritable bowel syndrome: a gut microbiota-related disorder?” American Journal of Physiology-Gastrointestinal and Liver Physiology, vol. 312, no. 1, pp. G52–G62, 2017, pMID: 27881403. [Online]. Available: https://doi.org/10.1152/ajpgi.00338.2016

[50] P. Buono, P. Tamla, T. Krause, F. De Luzi, F. Monti, M. Mecella, “Revisiting Data Visualizations in Diagnostic Reports,” in Proceedings of the 2025 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBM Workshops). Wuhan, China: IEEE, Dec. 2025, workshop on Genomic Foundation Models for Diagnostic Innovation (GFM4D 2025), co-located with IEEE BIBM 2025, to appear.