

## Article

# The Cloud-Enabled Architecture of the Clinical Data Repository in Poland

Dariusz R. Augustyn <sup>1</sup>, Łukasz Wycislik <sup>1,\*</sup> and Mateusz Sojka <sup>2</sup>

<sup>1</sup> Department of Applied Informatics, Faculty of Automatic Control, Electronics and Computer Sciences, Silesian University of Technology, 44-100 Gliwice, Poland; [dariusz.augustyn@polsl.pl](mailto:dariusz.augustyn@polsl.pl)

<sup>2</sup> Independent Researcher, 43-100 Tychy, Poland; [matisojka666@gmail.com](mailto:matisojka666@gmail.com)

\* Correspondence: [lukasz.wycislik@polsl.pl](mailto:lukasz.wycislik@polsl.pl)

**Abstract:** In this article, the authors, using information-systems modeling techniques, and considering current national legal regulations, present the cloud-enabled architecture of a clinical data repository. The patient's medical record is an important carrier of information necessary for accurate diagnosis and selection of the correct treatment process. Therefore, it is not surprising that since the beginning of the development of computer technologies, databases have been built to enable the management of a patient's medical records. These systems were most-often deployed locally at individual healthcare units, which carried certain limitations both in terms of the security and availability of the stored information, and the possibility of exchanging it with other clinics. However, recent developments in the standardization of medical information exchange in Poland, together with the revolution in cloud computing, have opened up completely new perspectives for clinical-data-repository implementations helping to make them far more sustainable. Although, the practical aspects of implementing clinical-documentation repositories are studied both in forums of European countries and also around the world; so far, no similar research was conducted with respect to Poland. This study tries to fill that gap by proposing a flexible multi-variant cloud-enabled architecture of the system providing the services of a clinical-data repository. The goal of the work was to propose such a system architecture that allows having a system that is either cloud-agnostic, that uses specifically selected cloud services, or that is even deployable locally. Thanks to the use of cloud computing services, the implemented system is characterized by high availability, scalability, and the possibility of exchanging data between medical institutions, which enables the improvement in the quality of medical processes for the whole Polish population.

**Keywords:** clinical documentation; repository service; cloud computing; containerization; kubernetes; high availability; relational database; object storage



**Citation:** Augustyn, D.R.; Wycislik, L.; Sojka, M. The Cloud-Enabled Architecture of the Clinical Data Repository in Poland. *Sustainability* **2021**, *13*, 14050. <http://doi.org/10.3390/su132414050>

Academic Editor: Ewa Ziemba

Received: 29 October 2021

Accepted: 11 December 2021

Published: 20 December 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In recent years, the pace of the advancement of society has been enhanced by uninterrupted growth in the spread of information and communication technologies (ICT) and in ICT uptake by citizens, enterprises, and public organizations, as well as the increasing role of information in all spheres of life [1]. A society that is good at employing ICT to create, disseminate, and utilize information effectively can gain social and economic advantages, reach the cutting edge of competitive markets, and pioneer new avenues for welfare creation for their citizens [2]. This is, in particular, true when it comes to the healthcare sector [3], where massive information describing one's state of health is being produced every single day. The proper optimization of clinical information workflow, exchange, and retention may contribute to sustainable development in terms of equalization of universal access to medical services, lowering the overall costs of both medical services being provided and the IT infrastructure needed to host clinical information systems, even reducing the footprint on the environment [4].

Computing clouds are of great importance for the effective implementation of IT systems at the level of regions or entire countries [5]. While in the case of government agencies with an appropriate budget, proprietary data-processing centers are often still used, commercial entities wishing to enter their IT solutions on large markets are increasingly turning to specialized cloud service providers whose offer includes not only the latest achievements of the IT world but is also available on a pay-as-you-go basis. As a result, the use of computing clouds has an extremely low entry threshold, and the costs incurred depend on the scale of use of individual services of a given cloud service provider.

Until recently, in Poland, the patient's medical records were kept in paper form (or partially in electronic form) and stored only locally, usually at the place where they were created, i.e., in individual health facilities. Recent organizational and legal changes in Poland have provided a good foundation for the full digitalization of the circulation of medical documents. Although, the practical aspects of implementing clinical documentation repositories are studied both in forums of European countries, such as Austria [6] and Norway [7], and also around the world, for example in Taiwan [8], Nigeria [9], and Pakistan [10], according to the best knowledge of authors, so far, no similar research was conducted with respect to Poland. This study tries to fill this gap by proposing a flexible cloud-enabled architecture of the system providing the services of a clinical-data repository. In this article, the authors discuss variants of cloud-enabled architecture of the clinical-data repository system, taking into account the Polish specificity of new legislation and organizational conditions, which, at the same time, are adapted to the deployment of the computing clouds. The goal of the work was to propose such a system architecture that allows having a system that is either cloud-agnostic, that uses specifically selected cloud services, or that is even deployable locally. Relevance of the architecture was confirmed by its wide usage by CDR systems for hundreds of medical clinics in Poland (SaaS solution provided via GCP for small medical facilities or on-premise applications deployed in hospitals).

What are the benefits of implementing a nationwide cloud-based clinical-documentation-repository system? What should be taken into account to maximize the potential benefits of migrating computing to the cloud for the system in question? How should we prepare for an unexpected increase in workload caused by a growing number of users? We tried to provide answers in this study.

The flow of content for this article is as follows. The Related Works section describes the current state of research on the standardization of medical-documentation reporting and the role of cloud computing in the implementation of systems for entire societies, in particular, in the health sector. The Materials and Methods section presents the organizational and legislative foundations in Poland on the basis of which it is possible to implement the nationwide clinical-data-repository system. It also includes a simplified description of the requirements for the software being built. The Results section presents in detail the proposed system architecture that meets the previously set requirements and that, at the same time, meets the criteria of the sustainable-development paradigms. The Discussion section points to the advantages of the proposed solution but also some threats and challenges that must be taken into account when deploying the proposed system in the cloud. Finally, the directions for further research are outlined too.

## 2. Related Works

The standards of medical procedures and medical diagnosis are constantly evolving to ensure the highest possible effectiveness of patient treatment [11]. This progress, benefiting entire communities, is enabled largely due to the development of the processing technology of big data sets describing clinical cases [12]. At the same time, however, the health and life of an individual may depend on the availability and accuracy [13] of medical data collected about him in various healthcare facilities. It should also be noted that, on the one hand, the desire to improve the quality of collected clinical data may result in activities that standardize the structure of electronic health records (EHR), but, on the other hand, its excessive formalization may slow down the development of medical techniques. Not

surprisingly, clinical-data storage and sharing implemented by clinical-data repositories (CDRs) is the subject of intense research in the combined fields of medicine, management, law, and computer science.

Seeking for an interoperability between individual hospital information systems (HISs) used in given clinics resulted in the development of the HL7 standard. HL7 stands for Health Level Seven, where level seven refers to the application layer of the ISO/OSI computer network model [14]. The origins of this standard date back to 1987, and the organization under the same name (<https://www.hl7.org/> (accessed on 16 August 2021)) is responsible for its development. The aim of this organization is to develop standards for the electronic exchange of clinical, financial, and administrative information between IT systems in healthcare. The standard, of course, has evolved over time, adapting more and more new achievements of IT technology. The current version 3 of the standard has completely changed the communication model. In this version, the standard is based on the XML language, which thus makes it easy to use the object model.

The main purpose of the standard is to ensure proper communication between medical systems. Examples of medical systems include the Radiology Information System (RIS), the Lab Information System (LIS), Hospital Information System (HIS). Each system has a different specificity and functionality. The main task of HL7 is to facilitate the implementation of interfaces for these systems and reduce the cost of their development, so that the exchange of data between systems is cheaper and less complicated. The standard also intends to achieve the following goals:

- Implementation of data exchange between systems based on various protocols;
- Support specification development when new requirements are set;
- Use of existing standards related to the standard. It should not favor the interests of specific companies;
- The long-term goal should be to define formats and protocols for computer programs in all healthcare facilities.

The HL7 standard is sometimes referred to as the “nonstandard standard.” This formulation emphasizes the flexible nature of the standard, which describes the method of communication between systems in a general manner and not related to a specific IT technology, programming language, etc. HL7 is a very popular standard in the United States. It is used in over 90% of medical facilities in this country. This standard is also being developed in other countries.

Initially, the standard did not meet with much interest from vendors of medical IT systems. Members of the HL7 organization noted, however, that it is not necessary to implement the standard 100% to reduce significantly the cost of interface development. It turned out that it is enough to define only 80% of the system interface in accordance with the standard, leaving 20% of the interface for other functions specific to a given system [15]. This approach contributed to the increased interest in the standard by companies producing medical solutions.

The standard does not define the implementation details of medical systems. Two systems based on and fully compatible with HL7 may not be compatible. The solution to this problem may be special interface engines created by HL7, which act as intermediaries between the systems.

Most of the HL7 standards are considered open standards. Since April 2013, they are available to everyone for free, and it is enough to register on the organization’s website. Today, HL7 seems to be the most mature proposal for the exchange of medical information, but even the best concepts will not work properly without their implementation and deployment based on a reliable runtime infrastructure.

It is commonly believed that specialization drives any technical and civilization progress. Cloud computing is being specialized toward providing remote computing resources with the possibility of rapid provisioning [16], and thus it is often compared, when it comes to the importance and contributions to the development of mankind, to the introduction of the first power plants. Thanks to cloud service providers, users need

not worry anymore about hardware devices and the means (like electricity, computer networking, and cooling systems) necessary to maintain all aspects of infrastructure, likewise formerly, consumers of electricity did not need to worry about fuel, the maintenance of power generators, and so on. Despite the term “cloud computing” becoming popular in 2006 when Amazon [17] started offering its elastic compute cloud (EC2) services, it was in fact coined ten years earlier by Compaq [18].

The beginnings of cloud computing were much more modest than what can be found in the offers of many vendors today. First, the virtual private server (VPS) (a.k.a., a virtual machine, or VM for short) services have been developed along with all the supporting features like access keys management, block storage devices, VM snapshots, etc. These services form the basic and most generic layer (called Infrastructure as a Service—IaaS) of the whole cloud computing stack, on top of which the more abstract layers are being built, e.g., platform as a service (PaaS), software as a service (SaaS), and many, many more [19]. Nowadays, thanks to this concept, the whole range of services and products are made available from the cloud, including but not limited to databases (both SQL and NoSQL), business applications, business intelligence, and e-commerce platforms, or even quantum computing and block-chain systems.

The promises of cloud service providers to increase fault tolerance, increase security, increase efficiency, and more while reducing the costs of operating IT infrastructure have naturally attracted the attention of not only commercial but also scientific entities.

A significant amount of scientific interest is devoted to the issues of the security of computing clouds both in terms of services made available from the cloud data centers [20] and in terms of data stored in the clouds, especially in the case of multitenant systems [21].

As the computing clouds (more precisely, cloud computing data-centers) have large reserves of computing power, the advantages of using their services stem not only from the completeness and richness of the services provided but also from computing, data-storing redundancy possibilities, and the potential scaling, understood as keeping certain time metrics related to response times, to the growing number of concurrent users or tasks. Of course, the need for redundancy and scaling is for mission-critical applications, and CDRs are certainly one of those.

Research on the use of cloud computing in the field of medicine is very extensive both in terms of its application [22,23] and security [24]. What cloud computing can bring to healthcare depends on how it will be utilized. Correctly implementing and utilizing cloud computing will offer a healthcare organization practice enormous benefits [25]. The evidence of the rapid and pressing technological evolution—joint with the worldwide rising governmental efforts—leads to the conclusion that the healthcare sector is already facing the impact of I4.0, effectively moving eHealth towards Healthcare 4.0 [26].

In the case of a clinical-data repository, the issues of cloud-based data stores and the method of providing services via network interfaces are particularly important. Storage space plays a vital role while dealing with the increasing amount of data. Many laboratories and hospitals may suffer the lack of appropriate storage space [27]. That is why cloud solutions can be especially attractive for smaller clinics and medical practices that have not yet built their own data processing centers. Additionally, large clinics are most often unable to provide data stored with the level of data protection and availability offered by cloud service providers. On the other hand, the availability and popularity of Web services supported by the development of associated standards, protocols, and technologies, like XML, SOAP, WSDL and UDDI, provided the opportunity for building complex processes that are based on the message flow between many distributed web services. However, the construction of such systems requires the appropriate coordination of all components taking part in the composite process [28]. IT systems for the healthcare sector are complex ecosystems of services cooperating to support the implementation of health processes. Many global standards in the field of health care successfully use communication using network services, in particular in the area of clinical-data repositories.

It is worth noting that while clinical-data repositories may contain data similar to biorepositories, the method to search for information will usually be different. In the former, the data are searched for in the context of a given patient; in the latter, the data are arranged differently. The existing biorepository landscape is highly heterogeneous and populated with individual laboratory collections, private collections, organism-specific collections, government collections, select-agent collections, user-group collections, geographical collections, and large-scale public or commercial collections [29]. Additionally, expected non-functional requirements will usually differ—in the case of the former one, the most important will be availability, security, and accountability, while biorepositories are expected to be usually easily accessible and concise in their format.

The use of cloud technologies focused specifically to improve the management of clinical records has been studied for several years worldwide. A good example of that is a study [30] aiming to introduce and describe an architectural solution that can extend the classic EHR to support new monitoring and managing features by making full use of the possibilities offered by cloud computing and big data analytics in which an attempt of coining a new acronym of cloud-based electronic health record (CEHR) was made. An interesting approach is presented in the article [31] describing the implementation of personal health record system developed using cloud technologies to allow constant monitoring capability by supporting dynamic creation of clinical document architecture (CDA) from a mobile device. The generated CDA document may be used to assess current health against major diseases through a clinical decision support system. Knaup et al. [32] emphasized the importance of interoperability for building global repositories of clinical documentation. It also reviews the basic standards of document exchange, which have already gained popularity. In yet another article [33], the authors note the benefits of depositing medical documents using cloud computing services. They propose an approach that manages personal health data by utilizing meta-data for organization and easy retrieval of clinical data and cloud storage for easy access and sharing with caregivers to implement the continuity of care and evidence-based treatment. In case of emergency, the solution they propose makes critical medical information such as current medication and allergies available to relevant caregivers with valid license numbers only. Additionally, studies can be found that are less theoretical and that focus on the analysis and architecture design of the practical implementation of clinical documentation repositories in individual countries, for example, for Taiwan [8], Austria [6], Norway [7], and more. Developing countries, such as Nigeria [9] or Pakistan [10], also see great benefits from implementing similar solutions and are actively engaged in research in this direction.

### 3. Materials and Methods

Recently, good foundations for the digitization of medical processes have also been created in Poland, but according to the best knowledge of authors, so far, no practical attempt of the CDR system implementation in that country has been described. This research fills that gap.

#### 3.1. Legal and Organizational Status in Poland

The processes of ICT adoption by the healthcare system in Poland began as early as 1993 but initially included only the processes of contracting, settlement of the services provided, and verification of the patient's entitlement to obtain medical services. The first attempts to implement a national register of medical services took place in 1996, but they failed, and the situation did not improve for many years. On 15 April 2011, the Act on Medical Activity [34] was passed in Poland, which, along with Regulation of the Minister of Health of 21 December 2010 on the types and scope of medical documentation and the method of its processing [35], provided a legal framework for improving the organization and computerization of the healthcare sector in Poland. However, it was only in 2014, when the adoption of the HL7 protocol actually began, that it became possible to develop standards enabling the implementation of clinical-documentation repositories that store

documents in a form in which they can be exchanged with other repositories and their content properly interpreted.

In January 2017, the Polish HL7 Association (<https://hl7.org.pl/> (accessed on 16 August 2021)) started its activity, which—on the basis of an affiliate agreement signed with HL7 International—is the official national organization of HL7 in Poland. In international contacts, the association acts, in accordance with the convention adopted in HL7, under the name HL7 Poland. The members of the organization are natural persons (individual membership) or companies and other entities with legal capacity.

In accordance with the statute adopted by the founders, the goals of the association are as follows:

- Cooperation with HL7 International, on the terms set out in the affiliate agreement, granting the association the rights and obligations of the HL7 national organization;
- Development and commissioning of technical specifications and other materials related to HL7 standards, in particular, translations and national localizations and other HL7 materials, within the meaning of the affiliate agreement concluded with HL7 International;
- Providing assistance to entities undertaking activities consisting in creating their own locations of HL7 standards and their correct implementation and application;
- Supervising the correct implementation and application of HL7 standards, their location, and other materials related to HL7 standards and respecting the intellectual property rights of HL7 International and other entities;
- Organizing, conducting, and participating in projects aimed at increasing interoperability in healthcare;
- Promotion of interoperability standards in healthcare, in particular, organizing meetings, conferences, seminars, trainings, examinations, and distribution of information and educational materials on this subject.

Today, the developed standard [36] is already in use and even is required by law. It is called HL7 PIK (Polska Implementacja Krajowa, which means Polish National Implementation). For the issues of the clinical document repository, the most important is its subset clinical document architecture (CDA), which standardizes the structure of medical documents. However, the CDA does not describe how the documents are to be transmitted; therefore, in Poland, the IHE standard (<https://www.ihe.net/> (accessed on 16 August 2021)) is additionally used for processing transactions based on IHE IT Infrastructure (ITI) specification [37], according to which the services are exposed as SOAP endpoints [38] implementing XDS.b profile [39] and exchanging documents between XDS.b-document-repository and XDS.b-document-source actors. A typical operation performed by the CDRs is, for example, ITI-41 for providing and registering documents of Set-b, but there are also many, many more [40].

### 3.2. System Requirements

The requirements for the system presented in this article have been defined at a fairly high level of abstraction, as most of them are already described by Polish legislation or industry standards. The most-important part of the considerations presented in this research is how cloud environments can support the implementation of these requirements so that the system is secure, easily scalable, and at the same time cheap to maintain and operate for end-customers, offering countrywide access to the clinical documents and reducing the environment pollution footprint by moving the localization of computation from individual distributed server rooms to cloud data-centers optimized in terms of electricity consumption per unit of computing power.

Therefore, the problems that the proposed architecture solves can be presented in the following few bullets:

- Supporting common industry standard of security;
- Supporting national (e.g., PIK HL7 CDA) and general clinical-documentation-exchange standards (e.g., HL7 CDA, IHE, and XDS.b);

- Providing a flexible multi-tenancy model;
- Satisfying HA in the fault-tolerant aspect;
- Satisfying HA in the efficiency aspect;
- Supporting cloud agnosticism;
- Supporting compatibility to on-premise solutions (preserving ability to local deployment);
- Supporting various storage providers for persisted clinical documentation;
- Supporting methods of monitoring and tracing.

#### 4. Results

The proposed general architecture that satisfies the requirements mentioned in Section 3.2 is shown in Figure 1 based on the component blocks diagram.

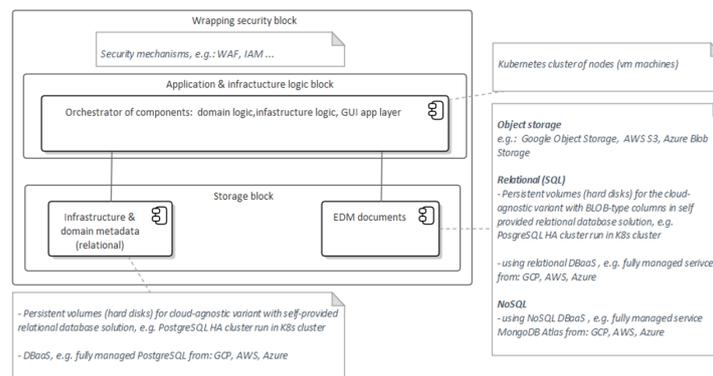


Figure 1. Basic architectural blocks.

The entire solution should be wrapped/supported by standard security mechanisms like: a web application firewall (WAF), identity and access management (IAM), and many others. This wrapping security block is the outermost layer (cloud one) of the general concept of the 4Cs security model [41] (Figure 2)—cloud, cluster, container, and code, whose detailed description and implementation is outside the scope of this study.

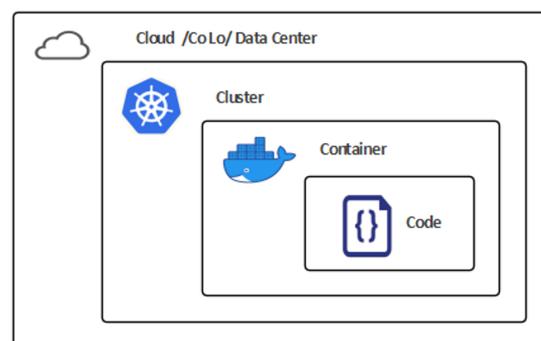


Figure 2. 4Cs Security Model— Cloud , Cluster, Container, Code.

The application and infrastructure block consists of components providing:

- *Domain logic* connected with functionality of storing, retrieving, and processing clinical documents,
- *Infrastructure logic*—functionality of authentication/authorization, tenant management, security configuration, and request and error monitoring/tracing,
- *Self-provided system functionality* like a relational database server, e.g., PostgreSQL high-availability cluster, or a memory cache NoSQL database, e.g., Redis Cluster.

This block also delivers GUI parts of web applications (the block serves a static content of a single-page application (SPA) like HTML, CSS, and Javascript files). Due to properties like high availability, scaling granularity, and autoscaling, this block is implemented using

some orchestrator. Kubernetes (K8s) [42] is currently the most-popular orchestrator for the systems based on containerized microservices. This trend is observable not only in commercial systems for clinical data processing but also in scientific studies (see [43]).

Most of the modern solutions are based on the microservice architecture where a piece of application logic is provided by a single ready-to-use component that exposes its functionality via a public network interface (commonly, HTTP RESTful services but also SOAP web services or an asynchronous queue/topic mechanism like JMS). Modern methods of deployment assume the usage of containerization. A container is an instance of a container image that consists of an OS layer and an appropriate set of runtime libraries and a proper application that could be implemented as a microservice. An entire system is a set of independent and isolated but cooperating containers. These properties of the microservice architecture and containerization allow either to develop or to deploy microservices independently.

Containerization has been popular starting from the Docker [44] environment general availability (especially, in on-premise solutions but also in cloud-based ones). To satisfy high availability (in a sense of fault tolerance and efficiency), the architecture assumes the use of orchestrator like Kubernetes. This allows to run a whole solution on a cluster of computers/virtual machines called nodes. A K8s cluster can be deployed locally in an on-premise system or it can be provided as a cloud-managed service like the Google Kubernetes Engine (GKE) in GCP or the Elastic Kubernetes Service (EKS) in AWS or the Azure Kubernetes Services (AKS) in Microsoft Azure. In the K8s, containers (commonly the only one) are embedded in pods (Figure 3). A pod is the smallest unit of processing and can be run on any node. The master K8s module (also called a control plane) manages pod instances and their distribution among the nodes. The has a special kind of K8s resources—deployment is responsible for grouping pods and setting of pod parameters like: the source container image, the desired number of pods, the endpoints for checking pod readiness and liveness, the thresholds for requested CPU and RAM usage, etc. Another kind of K8s resource is that the service is responsible for distributing (and load balancing) requests among a set of pods defined by some deployment. Services can be used either to communicate inside a K8s cluster (ClusterIP services) or to expose some services outside (NodePort and LoadBalancer ones). HPA (horizontal pod autoscaler) is a K8s resource responsible for automatic increasing or decreasing the number of pods depending on given threshold values defined on mean CPU usage for a selected set of pods. All resources mentioned above can be grouped into namespaces, which introduces an additional granulation level which is useful for management of parts of a K8s cluster. Resources like a persistent volume claim (PVC) and a persistent volume (PV) allow to allocate and manage volumes needed to be persistent across the K8s lifetime (hard disk storage). They allow to provision volumes for pods where storage space is delivered from outside the K8s cluster (e.g., it may come from local hard disks for an on-premise system or NFS servers or disk space services from a particular cloud provider).

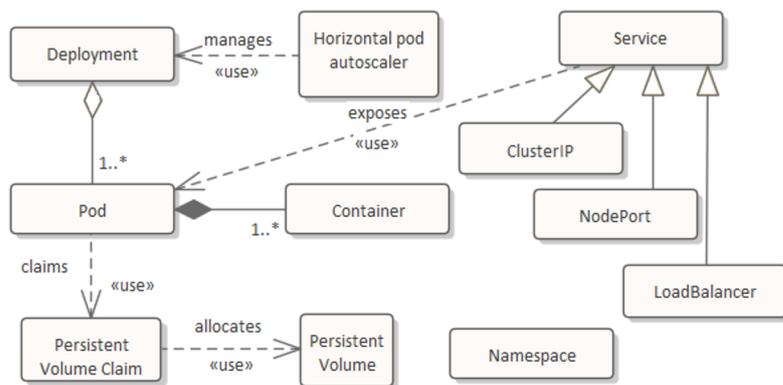
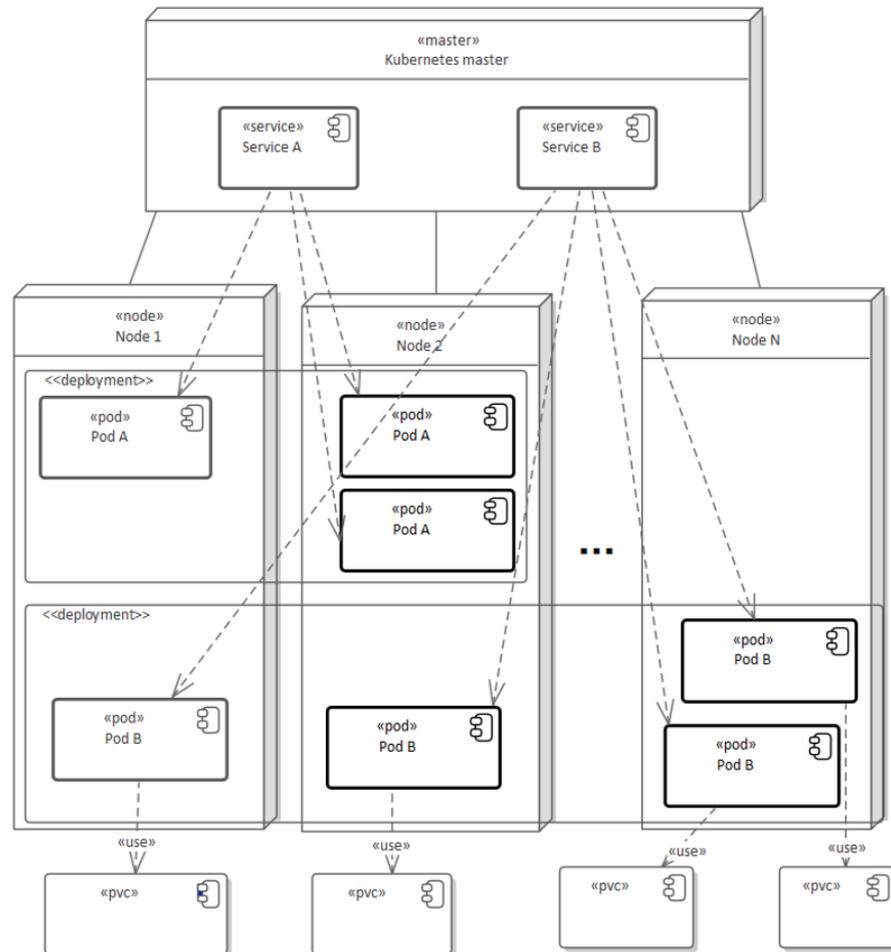


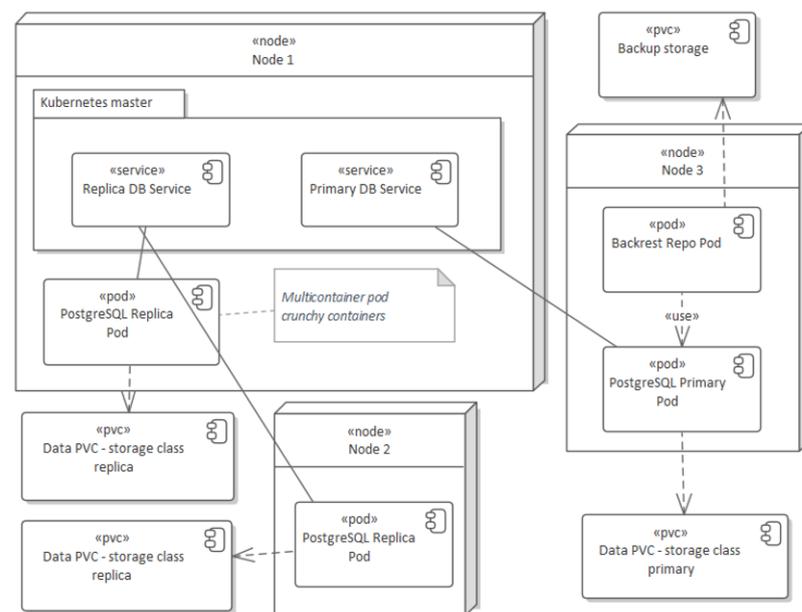
Figure 3. Selected basic types of Kubernetes resources.

Figure 4 presents the use of basic K8s resources in construction of a simple cluster that consists of two services and two deployments. Pods (A or B) represent typical execution units providing domain services that are stateless and easily horizontally scalable (easily multipliable and allocated on many nodes).



**Figure 4.** Sample diagram of usage of K8s resources (node, deployment, pod, PVC, and service) in deployment context.

Since PVC and PV are used for allocating volumes needed for a pod, they are also used for allocating storage needed for pods belonging to the variant of applied solution that supports PostgreSQL HA cluster—Crunchy PostgreSQL for Kubernetes [45]. The above-mentioned solution allows configuring one master pod where SQL inserts, updates, and deletes may be performed, and many PostgreSQL replica pods where data are replicated from the master pod and all SQL selects may be distributed among all replicas. Figure 5 presents a sample three-nodes K8s cluster with one progress master pod and two replica ones. A crunchy data PostgreSQL HA cluster supports not only horizontal scaling of replicas but it allows to eliminate the risk of master failures by the mechanism of automatic switching the role from the replica pod to the master one. Crunchy PostgreSQL also provides a functionality of making a database backup (backrest repo pod) using data handled by a master PostgreSQL Pod.



**Figure 5.** Deployment of crunchy high-availability PostgreSQL cluster run in a three-nodes Kubernetes cluster.

The specific nature of clinical documents (PIK HL7 CDA level 3 in Poland), i.e., XML files, makes it that they can be stored using various methods. The most basic approach relies on storing them in BLOB-type columns in a relational database (RDB), taking into account that RDB may be used for storing and processing medical metadata related to medical documents, and the same RDB is used for data of the infrastructure part of a system, too. Such an approach is commonly used in on-premise solutions, but it is also possible in cloud ones. However, cloud providers extend possibilities in this area in aspects of high availability and lowering costs by introducing specialized managed storage services. One of them is object storage—the service for storing unstructured files in buckets. Object storage is significantly cheaper than using block devices (hard disks storage) consumed by RDB. Additionally, we may select an appropriate cost plan by selecting a class of storage (depending of number of reads vs. writes), which allows to tailor the solution to the nature of CDR usage. Increasing the availability and the fault tolerance of object storage is easy by using multi-regional option. To support possibility of scaling operations on documents and their metadata the described system is also prepared for working with a document-oriented database, i.e., MongoDB. This option was introduced to support the ability of full horizontal scaling of NoSQL server with many identical workers (which is not available in RDB HA cluster with only one master for writing and many replicas for reading). Summarizing, the CDR system supports many clinical-document=storage providers: RDB (PostgreSQL: self-provided PostgreSQL HA cluster—crunchy data or PostgreSQL DBaaS for GCP, AWS, Azure) or NoSQL (MongoDB self-provided or DBaaS MongoDB Atlas for GCP, AWS, Azure), or the most-promising object storage (cloud storage in GCP, S3 in AWS, Azure Blob Storage in Microsoft Azure).

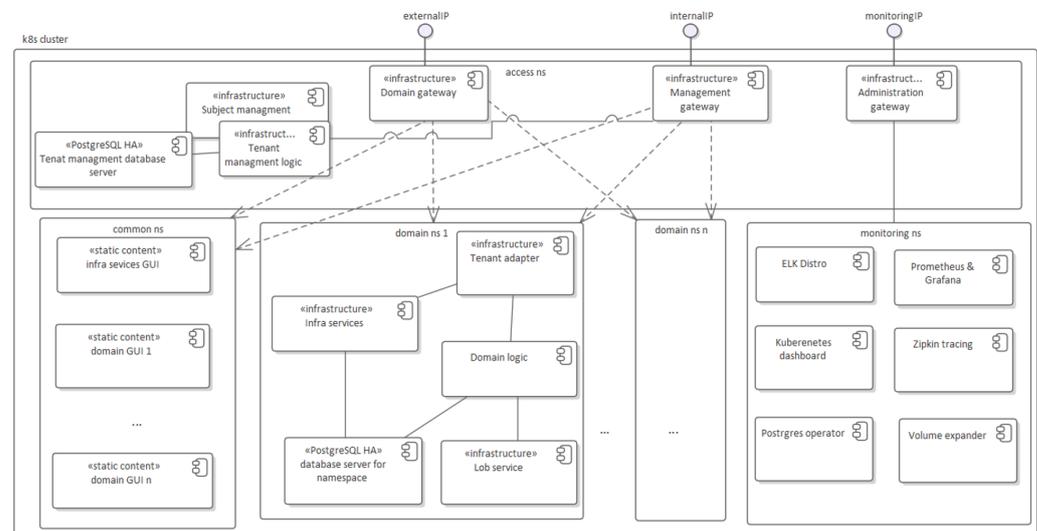
The system based on the described architecture was deployed in the Google Cloud Platform and run in production with:

- RDB for infrastructure data and metadata (self-provided)—PostgreSQL HA cluster—crunchy data;
- Google Cloud Storage for clinical documents.

Figure 6 shows the detailed architecture of the application and the infrastructure block (from Figure 1) according to the K8s concept. There are three access externalized endpoints:

- *ExternalIP* for accessing client/subject to domain exposed services;
- *InternalIP* for configuring and administering access by client tenants;

- *MonitoringIP* for monitoring, tracing, and database administering. Providing those three endpoints allows applying customized different appropriate security mechanisms to each one.



**Figure 6.** Kubernetes cluster of clinical document repository system.

The application and infrastructure block is based on separate namespaces (Figure 6):

- *Access ns*—with pod providing the logic of tenant administering and accessing the K8s cluster;
- *Common ns*—with pod delivering static contents like GUI;
- *Monitoring ns*—with modules for: administrating K8s cluster (Kubernetes dashboard [46]), central viewing of merged error logs (ELK stack [47])—Elasticsearch, Logstash, Kibana), monitoring state, utilization, and efficiency of K8s cluster (Prometheus and Grafana [48]), detail tracing request inside K8s cluster (Zipkin [49]);
- *domain ns 1 . . . domain ns n*—with pods providing the proper domain logic of creating and processing PIK HL7 CDA documents (and some pods providing the necessary infrastructure logic).

Figure 6 shows some important deployments (set with at least one pod) placed in an appropriate namespace. Some deployments play special roles. For example, the component denoted by PostgreSQL HA is a simplified notation of the solution presented in Figure 5. The deployment lob (large object) is responsible for storing/retrieving documents, i.e., it implements storage providers like the above-mentioned PostgreSQL, object storage, MongoDB, etc. Each subject (remote client of CDR Repository system) has its own tenant in the described system. Tenants are grouped and handled by deployments in one of the domain namespaces. This means that a pod works for all tenants that belong to the pod's namespace. Such an approach allows achieving a convenient level of granulation (tenant level and namespace one). Thus, for example, we may increase the number of handled tenants by pods inside the current namespace, and when we achieve too high of a CPU utilization of pods in the namespace, we may create another namespace for new-coming tenants.

## 5. Discussion

The adoption of cloud computing along with progressive standardization brings unprecedented possibilities of improvements in clinical-data management, which in turn leads to the rising quality of medical processes. For some time, Polish legislation in the areas of public healthcare has been systematically adjusted to the global trends in the digitalization of this branch, which makes it possible to implement more and more modern and sustainable IT solutions. These directions also seem to be supported in general by

theoretical models of ICT adoption contribution to sustainability in Poland and are created by scientists in relation to households [50] or even local governments [51]. The system architecture proposed in this study may contribute to sustainability enhancement within economical, environmental, and social equity dimensions compared to legacy solutions being used so far in Poland.

In the case of data repositories, and such are undoubtedly clinical data repositories, the data storage layer is a crucial component. The proposed solution, thanks to the development of dedicated interfaces, enables the use of many popular data storage solutions that can be used in computing clouds, both in the managed model (i.e., “database/storage as a service”) and in the BYO model (i.e., “bring your own”). Moreover, the separation of the data storage layer into the document metadata store and the document content store provides additional flexibility to manage the costs of operating the system. By using managed solutions, the system administrator/operator can easily use the offer of individual cloud computing providers, enabling easy configuration of the data storage layer in areas such as availability, redundancy, scaling, etc. However, by implementing them by his own basing on the most generic services (such as virtual block devices), he gains the possibility of full control over the configuration but at the cost of additional effort for the deployment and maintenance of the system. The use of various data-storage technologies allows not only to adapt to various functional requirements but also allows one to optimize costs depending on the volumetric characteristics of the system usage by individual customers, such as the number of documents stored per unit of time, the number of document reads per unit of time, or the retention time of documents.

The choice of using PostgreSQL as a metadata store was influenced, on the one hand, by its great popularity in the group of open source solutions, and, on the other hand, by its very high maturity and extensibility. At the same time, these features make this database management system widely adapted by many cloud service providers and made it available in the DBaaS model. This cannot be said for the case of commercial platforms such as Oracle or MS SQL Server, where the popularity of this adoption is hampered both by the need to incur license costs but also by greater complexity when it comes to deployment and configuration on cloud resources.

By using the popular S3 service as a data storage layer for clinical documents’ content, it allows, for example, for most cloud service providers to have a very good cost optimization thanks to the appropriate selection of the S3 service class in terms of availability and characteristics of reads and writes—for example, AWS, apart from the standard S3 services, also provides a very cheap version of Glacier and even Deep Archive Glacier [52]. The attractiveness of the price of this service results from the applied technical storage means, which could be traditional and cheap tape libraries (although AWS does not disclose in the documentation what specific technology was used for implementation) having a large capacity, but the disadvantage is that the process of reading (or more precisely retrieving) the data stored on them is long. Usually, the offer is structured in such a way that saving data using such a service is very cheap, while reading it takes a significant time and is more expensive. The advantage of using S3 services can also be its standard object access protocol [53], thanks to which files can be read not only by dedicated software but also by popular standalone applications that are S3 clients (S3 browsers).

On the other hand, the use of an SQL database as a document storage layer makes it easier to obtain the consistency of the content of documents with the metadata describing them, because then a single component of the entire system is responsible for transaction management. In addition, thanks to the SQL language, we can build complex analytical queries, also those that may relate to the content of stored documents, if such a requirement would appear in the future. Then, the choice of the PostgreSQL database would be justified again, because this platform has the appropriate SQL extensions for processing XML documents, [54] and here it is in no way inferior to commercial solutions such as Oracle or MS SQL Server.

The use of a document-oriented database (i.e., MongoDB) as a document content storage layer also can have advantages at times as it would allow for very easy scaling of this layer (which is characteristic of NoSQL databases), while still maintaining the possibility of accessing documents in analytical queries (using MongoDB Query Language). However, it should be noted here that MongoDB does not natively support XML, so the documents would have to be converted to JSON format first.

In the proposed system architecture, the service access layer was implemented based on the concept of microservices, which, thanks to the usage of the K8s platform, made it relatively easy to ensure the availability and scalability of the system even in the case of multitenant installations. Sharing computing resources between multiple tenants has brought the expected significant savings of expenditures (TCO) but, at the same time, introduced some challenges in per tenant system monitoring.

The described concept of architecture was used in CDR system that provides services for KOChM (Krajowy Operator Chmury Medycznej—Operator of the National Medical Cloud) <https://chmuradlazdrowia.pl/> (accessed on 16 August 2021). It enables services in the Google Cloud Platform for hundreds of small medical facilities in Poland. The system was designed and developed by Asseco Poland S.A. company. All authors of the study played the role of architect of the solution.

The issue that needs to be mentioned is the legal regulations related to the processing of sensitive data, and such are undoubtedly the data contained in medical records. Polish law respects the GDPR European Union regulations [55] that require the processing of this type of data within the member states' borders. The assumptions of the system, the architecture of which is described in this article, go even further—all data being collected as part of this CDR is processed in Poland, specifically using the Google Cloud Platform data center located in Warsaw (*europa-central2* location [56]).

The implementation of the CDR system on a national scale, giving immediate access to complete data on the patient's health, chronic diseases or allergies, is a powerful tool supporting the work of doctors and allowing them to improve treatment processes. As it is now possible in Poland to issue electronic sick leaves and electronic prescriptions, in some justified cases (e.g., continuation of the treatment process of a given ailment), a "medical visit" may be carried out completely remotely, and thanks to the support of the CDR system described in this study, the doctor may make decisions based on complete and up-to-date information.

Of course, the implementation of a system for storing and sharing sensitive health data carries the risk of data leakage and its unauthorized usage. Therefore, it is important to apply the highest standards of security and protection of this data. If the data are collected within one medical unit, the technical responsibility lies with the provider of the software solution, and the organizational responsibility lies with the staff of this unit. The situation becomes significantly more complicated when medical records cross the borders of the units in which they were created, and this may often be the case in telemedicine systems or in cases of emergencies. In the first case, the patient's consent to transfer the documents to another medical unit should be recorded, and in the second, at least the reason for the transfer and the recipient of the document should be recorded. Information on the facts of disclosure of medical-records data should be credible and undeniable; therefore, even systems based on blockchain technologies are proposed for their processing [57].

Thanks to the proposed architecture involving concepts of microservices and containerization, the services may be deployed as a single node on-premise hospitals' own infrastructure or may be enabled as a cloud service in a SaaS model run on the K8s platform. In the latter deployment option, a high degree of cloud-agnosticism was achieved thanks to the technology stack composed of generic software components possible to deploy as BYO parts, e.g., ELK for logs monitoring, Zipkin Sleuth for requests tracking, Redis Cluster Sentinel for caching, or PostgreSQL Cluster for clinical document meta-data storage. However, using individual provider-specific cloud security layers, it is possible to achieve the highest security level of provided services.

In instances of systems deployed to production, there are significant daily spikes in the demand for computing power, with increasing intensity in the evenings. This presents some challenges in scaling a Kubernetes cluster horizontally, specifically in the provisioning and deprovisioning of compute nodes. Therefore, further research directions will focus on better scaling of the computing power of the system, either by developing predictive components that scale the K8s cluster, or by using serverless computing. A good example of using the latter approach would be a stateless data validation and XML document generation service in the PIK HL7 CDA level 3 format.

**Author Contributions:** Conceptualization, D.R.A., Ł.W. and M.S.; methodology, D.R.A. and Ł.W.; software, D.R.A., Ł.W. and M.S.; validation, D.R.A.; writing—original draft preparation, D.R.A. and Ł.W.; writing—review and editing, D.R.A. and Ł.W.; visualization, D.R.A. and M.S.; funding acquisition, D.R.A. and Ł.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by Statutory Research funds of Department of Applied Informatics, Silesian University of Technology, Gliwice, Poland (02/100/BK\_21/0008).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

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

## Abbreviations

The following abbreviations are used in this manuscript:

AWS	Amazon Web Services
BYO	Bring Your Own
CDA	Clinical Document Architecture
DBaaS	Database as a Service
CDR	Clinical Data Repository
CSS	Cascading Style Sheets
CEHR	Cloud-based Electronic Health Record
EC2	Elastic Compute Cloud
EHR	Electronic Health Record
GCP	Google Cloud Platform
HA	High Availability
HIS	Hospital Information System
HL7	Health Level Seven
IaaS	Infrastructure as a Service
IAM	Identity and Access Management
ICT	Information and Communication Technologies
IHE	Integrating the Health Enterprise
JMS	Java Message Service
JSON	JavaScript Object Notation
K8s	Kubernetes
LIS	Lab Information System
NFS	Network File System
OS	Operating System
PaaS	Platform as a Service
RIS	Radiology Information System
S3	Simple Storage Service
SaaS	Software as a Service
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
TCO	Total Cost of Ownership

VM	Virtual Machine
VPS	Virtual Private Server
WAF	Web Application Firewall
XDS	Cross-Enterprise Document Sharing
XML	Extensible Markup Language

## References

- Gries, T.; Grundmann, R.; Palnau, I.; Redlin, M. Innovations, growth and participation in advanced economies—A review of major concepts and findings. *Int. Econ. Econ. Policy* **2017**, *14*, 293–351. [\[CrossRef\]](#)
- Ziamba, E. The Contribution of ICT Adoption to the Sustainable Information Society. *J. Comput. Inf. Syst.* **2019**, *59*, 116–126. [\[CrossRef\]](#)
- Haluza, D.; Jungwirth, D. ICT and the future of health care: aspects of health promotion. *Int. J. Med. Inform.* **2015**, *84*, 48–57. [\[CrossRef\]](#)
- Sherman, J.D.; Thiel, C.; MacNeill, A.; Eckelman, M.J.; Dubrow, R.; Hopf, H.; Lagasse, R.; Bialowitz, J.; Costello, A.; Forbes, M.; et al. The Green Print: Advancement of Environmental Sustainability in Healthcare. *Resour. Conserv. Recycl.* **2020**, *161*, 104882.
- Dhar, S. From outsourcing to Cloud computing: Evolution of IT services. In Proceedings of the First International Technology Management Conference, San Jose, CA, USA, 27–30 June 2011; pp. 434–438. [\[CrossRef\]](#)
- Stolba, N.; Schanner, A. eHEALTH INTEGRATOR—CLINICAL DATA INTEGRATION IN LOWER AUSTRIA. In Proceedings of the CIMED2007, Third International Conference on Computational Intelligence in Medicine and Healthcare, Plymouth, UK, 25–27 July 2007. [\[CrossRef\]](#)
- Hertzum, M.; Ellingsen, G. The implementation of an electronic health record: Comparing preparations for Epic in Norway with experiences from the UK and Denmark. *Int. J. Med. Inf.* **2019**, *129*, 312–317. [\[CrossRef\]](#)
- Wu, C.H.; Chiu, R.K.; Yeh, H.M.; Wang, D.W. Implementation of a cloud-based electronic medical record exchange system in compliance with the integrating healthcare enterprise’s cross-enterprise document sharing integration profile. *Int. J. Med. Inf.* **2017**, *107*, 30–39. [\[CrossRef\]](#)
- Okediran, O.O.; Sijuade, A.A.; Wahab, W.B.; Oladimeji, A.I. A Framework for a Cloud-Based Electronic Health Records System for Developing Countries. In Proceedings of the 2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), Istanbul, Turkey, 14–15 April 2020; pp. 1–5. [\[CrossRef\]](#)
- Sarwar, M.A.; Bashir, T.; Shahzad, O.; Abbas, A. Cloud-Based Architecture to Implement Electronic Health Record (EHR) System in Pakistan. *IT Prof.* **2019**, *21*, 49–54. [\[CrossRef\]](#)
- Boudoulas, K.D.; Triposkiadis, F.; Stefanadis, C.; Boudoulas, H. The endlessness evolution of medicine, continuous increase in life expectancy and constant role of the physician. *Hell. J. Cardiol.* **2017**, *58*, 322–330. [\[CrossRef\]](#)
- Mehta, N.; Pandit, A. Concurrence of big data analytics and healthcare: A systematic review. *Int. J. Med. Inform.* **2018**, *114*, 57–65. [\[CrossRef\]](#)
- Sako, Z.Z.; Karpathiou, V.; Wickramasinghe, N. Data Accuracy in mHealth. In *Contemporary Consumer Health Informatics*; Springer International Publishing: Cham, The Switzerland, 2016; pp. 379–397.
- Melendez, W.A.; Petersen, E.L. The upper layers of the ISO/OSI reference model (part I). *Comput. Stand. Interfaces* **1986**, *5*, 13–46. [\[CrossRef\]](#)
- HL7 101: A Beginner’s Guide. Available online: [https://www.fortherecordmag.com/archives/fttr\\_01082007p22.shtml](https://www.fortherecordmag.com/archives/fttr_01082007p22.shtml) (accessed on 16 August 2021).
- Mell, P.M.; Grance, T. SP 800-145. *The NIST Definition of Cloud Computing*; Technical Report, National Institute of Standards and Technology, U.S. Department of Commerce: Gaithersburg, MD, USA, 2011.
- Announcing Amazon Elastic Compute Cloud (Amazon EC2)—Beta. Available online: <https://aws.amazon.com/about-aws/whats-new/2006/08/24/announcing-amazon-elastic-compute-cloud-amazon-ec2-beta/> (accessed on 30 September 2020).
- Who Coined ‘Cloud Computing’? Available online: <https://www.technologyreview.com/2011/10/31/210557406/who-coined-cloud-computing/> (accessed on 16 August 2021).
- Kavis, M. *Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS)*; Wiley CIO Series; Wiley: Hoboken, NJ, USA, 2014. [\[CrossRef\]](#)
- Subramanian, N.; Jeyaraj, A. Recent security challenges in cloud computing. *Comput. Electr. Eng.* **2018**, *71*, 28–42. [\[CrossRef\]](#)
- Kumar, P.R.; Raj, P.H.; Jelciana, P. Exploring Data Security Issues and Solutions in Cloud Computing. *Proc. Comput. Sci.* **2018**, *125*, 691–697. [\[CrossRef\]](#)
- Ali, O.; Shrestha, A.; Soar, J.; Wamba, S.F. Cloud computing-enabled healthcare opportunities, issues, and applications: A systematic review. *Int. J. Inf. Manag.* **2018**, *43*, 146–158. [\[CrossRef\]](#)
- Dang, L.M.; Piran, M.J.; Han, D.; Min, K.; Moon, H. A Survey on Internet of Things and Cloud Computing for Healthcare. *Electronics* **2019**, *8*, 768. [\[CrossRef\]](#)
- Chenthara, S.; Ahmed, K.; Wang, H.; Whittaker, F. Security and Privacy-Preserving Challenges of e-Health Solutions in Cloud Computing. *IEEE Access* **2019**, *7*, 74361–74382. [\[CrossRef\]](#)
- Wang, D. What will cloud computing bring to healthcare?—Overview of cloud applications in healthcare. *Int. J. Healthc. Technol. Manag.* **2015**, *15*, 49–57. [\[CrossRef\]](#)

26. Aceto, G.; Persico, V.; Pescapé, A. Industry 4.0 and Health: Internet of Things, Big Data, and Cloud Computing for Healthcare 4.0. *J. Ind. Inf. Integr.* **2020**, *18*, 100129. [CrossRef] [PubMed]
27. Mrozek, D. A review of Cloud computing technologies for comprehensive microRNA analyses. *Comput. Biol. Chem.* **2020**, *88*, 107365. [CrossRef]
28. Mrozek, D.; Małysiak-Mrozek, B.; Siążnik, A. search GenBank: Interactive orchestration and ad-hoc choreography of Web services in the exploration of the biomedical resources of the National Center For Biotechnology Information. *BMC Bioinform.* **2013**, *14*, 73.
29. Leighton, T.; Murch, R. Chapter 34—Biorepositories and Their Foundations—Microbial Forensic Considerations. In *Microbial Forensics*, 2nd ed.; Budowle, B., Schutzer, S.E., Breeze, R.G., Keim, P.S., Morse, S.A., Eds.; Academic Press: San Diego, CA, USA, 2011; pp. 581–601. [CrossRef]
30. Zangara, G.; Corso, P.P.; Cangemi, F.; Millonzi, F.; Collova, F.; Scarlatella, A. A cloud based architecture to support Electronic Health Record. *Stud. Health Technol. Inf.* **2014**, *207*, 380–389.
31. Song, Y.T.; Hong, S.; Pak, J. Empowering patients using cloud based personal health record system. In Proceedings of the 2015 IEEE/ACIS 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Takamatsu, Japan, 1–3 June 2015; pp. 1–6. [CrossRef]
32. Knaup, P.; Benning, N.H.; Seitz, M.W.; Eisenmann, U. eHealth and Clinical Documentation Systems. *Stud. Health Technol. Inf.* **2020**, *274*, 174–188.
33. Alyami, M.A.; Almotairi, M.; Aikins, L.; Yataco, A.R.; Song, Y.T. Managing personal health records using meta-data and cloud storage. In Proceedings of the 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), Wuhan, China, 24–26 May 2017; pp. 265–271. [CrossRef]
34. Ustawa z dnia 15 kwietnia 2011 r. o działalności leczniczej. Available online: <https://isap.sejm.gov.pl/isap.nsf/DocDetails.xsp?id=WDU20111120654> (accessed on 16 August 2021).
35. Rozporządzenie Ministra Zdrowia z dnia 21 grudnia 2010 r. w sprawie rodzajów i zakresu dokumentacji medycznej oraz sposobu jej przetwarzania. Available online: <https://isap.sejm.gov.pl/isap.nsf/DocDetails.xsp?id=wdu20102521697> (accessed on 16 August 2021).
36. Polska Implementacja Krajowa HL7 CDA. Available online: <https://www.cez.gov.pl/HL7POL-1.3.1.2/plcda-1.3.1.2/plcda-html-1.3.1.2/plcda-html-1.3.1.2/index.html> (accessed on 16 August 2021).
37. IT Infrastructure (ITI) Domain. Available online: <https://profiles.ihe.net/ITI/index.html> (accessed on 16 August 2021).
38. Latest SOAP Versions. Available online: <https://www.w3.org/TR/soap/> (accessed on 16 August 2021).
39. XDS.b Implementation. Available online: [https://wiki.ihe.net/index.php/XDS.b\\_Implementation](https://wiki.ihe.net/index.php/XDS.b_Implementation) (accessed on 16 August 2021).
40. ITI Vocabulary Registry and Data Dictionary. Available online: [https://wiki.ihe.net/index.php/ITI\\_Vocabulary\\_Registry\\_and\\_Data\\_Dictionary](https://wiki.ihe.net/index.php/ITI_Vocabulary_Registry_and_Data_Dictionary) (accessed on 16 August 2021).
41. Overview of Cloud Native Security. Available online: <https://kubernetes.io/docs/concepts/security/overview/> (accessed on 16 August 2021).
42. Production-Grade Container Orchestration. Available online: <https://kubernetes.io/> (accessed on 16 August 2021). [CrossRef] [PubMed]
43. Augustyn, D.R.; Wycislik, L.; Mrozek, D. Perspectives of using Cloud computing in integrative analysis of multi-omics data. *Briefings Funct. Genom.* **2021**, *20*, 198–206.
44. Developers Love Docker. Businesses Trust It. Available online: <https://docker.com/> (accessed on 16 August 2021).
45. Crunchy PostgreSQL for Kubernetes. Available online: <https://www.crunchydata.com/products/crunchy-postgresql-for-kubernetes/> (accessed on 16 August 2021).
46. Deploy and Access the Kubernetes Dashboard. Available online: <https://kubernetes.io/docs/tasks/access-application-cluster/web-ui-dashboard/> (accessed on 16 August 2021).
47. What is the ELK Stack? Why, it's the Elastic Stack. Available online: <https://www.elastic.co/what-is/elk-stack> (accessed on 16 August 2021).
48. Grafana support for Prometheus. Available online: <https://prometheus.io/docs/visualization/grafana/> (accessed on 16 August 2021).
49. Distributed Tracing with Spring Cloud Sleuth and Spring Cloud Zipkin. Available online: <https://spring.io/blog/2016/02/15/distributed-tracing-with-spring-cloud-sleuth-and-spring-cloud-zipkin> (accessed on 16 August 2021). [CrossRef]
50. Ziemba, E. The contribution of ICT adoption to sustainability: Households' perspective. *Inf. Technol. People* **2019**, *32*, 731–753. [CrossRef]
51. Ziemba, E. The Contribution of ICT Adoption by Local Governments to Sustainability—Empirical Evidence from Poland. *Inf. Syst. Manag.* **2021**, *38*, 116–134. [CrossRef]
52. Amazon S3 Glacier and S3 Glacier Deep Archive. Available online: <https://docs.aws.amazon.com/prescriptive-guidance/latest/backup-recovery/amazon-s3-glacier.html> (accessed on 16 August 2021).
53. Amazon S3 REST API Introduction. Available online: <https://docs.aws.amazon.com/AmazonS3/latest/API/Welcome.html> (accessed on 16 August 2021).
54. PostgreSQL 14.1 Documentation. Available online: <https://www.postgresql.org/docs/14/datatype-xml.html> (accessed on 16 August 2021).

- 
55. Complete guide to GDPR compliance. Available online: <https://gdpr.eu/> (accessed on 16 August 2021).
  56. Global Locations—Regions & Zones. Available online: <https://cloud.google.com/about/locations#regions> (accessed on 16 August 2021).
  57. Wyciślik, L.; Marcinkowska, E. Tracking of Clinical Documentation Based on the Blockchain Technology—A Polish Case Study. *Sustainability* **2020**, *12*, 9517. [[CrossRef](#)]