DECODE cloud platform: A new cloud platform to combat the burden of peripheral artery disease

Mohammed A. AboArab[1, 2]
[1]*Unit of Medical Technology and Intelligent Information Systems*
University of Ioannina
GR 45110, Ioannina, Greece
m.aboarab@uoi.gr
[2]*Electronics and Electrical Communication Engineering Department*
Faculty of Engineering
Tanta, Egypt
mohammed_aboarab@f-eng.tanta.edu.eg

Nikola Petrović[1]
[1]*Unit of Medical Technology and Intelligent Information Systems*
University of Ioannina
GR 45110, Ioannina, Greece
n.petrovic@uoi.gr

Vassiliki T. Potsika[1]
[1]*Unit of Medical Technology and Intelligent Information Systems*
University of Ioannina
GR 45110, Ioannina, Greece
vpotsika@uoi.gr

Dimitrios I. Fotiadis[1, 3]
[1]*Unit of Medical Technology and Intelligent Information Systems*
University of Ioannina
GR 45110, Ioannina, Greece
fotiadis@uoi.gr
[3]*Biomedical Research Institute,*
Foundation for Research and Technology-Hellas
GR 45110, Ioannina, Greece
fotiadis@uoi.gr

***Abstract*— Peripheral artery disease (PAD) is one of the most common diseases worldwide, especially in Europe. According to the literature, about 202 million people suffer from PAD. Powerful computational tools have been developed to address the widespread research interest in the study of PAD from a technical point of view. However, the lack of resources, expertise, or time necessary to install and use command-line tools or to deal with large datasets is a major hurdle to overcome. Thus, fast, dependable, and powerful applications are required to help in the treatment of PAD. The aim of the DECODE project is to establish a drug-coated balloon (DCB) simulation and optimization system for the improved treatment of PAD. The DECODE platform will integrate the data module to upload/download data from the repository. The data processors can log into the platform to access the image data processing and reconstruction module to perform the following: filtering, volume rendering, segmentation, measurement, and comparison. Moreover, tools of computational modeling, and machine learning will be provided. Additionally, users working in drug materials can directly access the DCB's geometries and properties storage module to upload and update the platform with any new material for optimization purposes aiming to establish new knowledge in the treatment of PAD and enhance clinical decision-making.**

Keywords—Peripheral artery disease, Cloud platform, Django framework

I. INTRODUCTION

Peripheral artery disease (PAD) refers to a blockage in the arteries delivering blood to the legs. A typical limb pain or intermittent claudication are common symptoms of PAD, which limit mobility and lower people's quality of life. It has been reported that PAD affects people aged 25 years and above. Recent data indicated that about 230 million have this disease [1]. Powerful computational tools have been developed because of the PAD widespread interest nevertheless, the lack of resources, expertise, or time necessary to install and use command line tools or to deal with large datasets is a significant problem [2]. The way products and services are designed is always influenced by technological progress. Web development has shown extraordinary expansion in recent years.

As a result, fast, dependable, and robust apps are required. There are two main types of applications: native and web [3]. Native applications are platform-specific (Windows, IoS, and Android) and are created with specialized programming languages and Software Development Kits, whereas Web apps are platform-independent web pages that look and feel like native apps in many aspects. Web apps are typically created in HTML5 (Hypertext Markup Language) and run in a browser. Native apps have complete access to the device hardware, whereas web apps have limited access. The utilization of native mobile applications has increased in recent years. In terms of performance, dependability, and engagement, web apps lag native apps. Developing, maintaining, and distributing native apps for certain platforms require additional resources and effort. In addition, publishing native apps is a time-consuming procedure. To utilize the native app, users must first register on the appropriate store, then check their memory, download, install, and lastly open it to use it. For every step between a user's first connection and the beginning of use, an app loses about 20% of its users [4]. This is complicated and time-consuming. Because both the web and native platforms have limitations, a platform that combines the capabilities and experiences of native apps with the reach of the web was needed. Such a platform depends on progressive web application (PWA) techniques. PWAs are web-based user experiences that are dependable, fast, and engaging [5]. PWA combines the best of the web with the best of native apps in one platform. It was first introduced in 2016 in the Google developer conference in San Francisco. It solved most of the issues that web apps lacked. PWA appears and operates like a native app due to the use of Service Worker, Web App Manifest, App shell, and Push notification. As PWA is a web app, it can be utilized across all platforms, saving costs, and improving access [6].

Cloud computing is defined as a model for enabling ubiquitous, on-demand network access to a shared pool of

This work has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 956470, as part of the DECODE project.

979-8-3503-9958-5/22/$31.00 ©2022 IEEE.

configurable computing resources (e.g., networks, storage, services, applications, and servers) that can be rapidly provisioned and released with minimal management effort or service provider interaction [7]. Cloud users can access these resources on demand to develop, host, and operate applications and services at any time, from any location, and on any device. The three service models used by a Cloud Service Provider (CSP) are Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). The four deployment methods of cloud computing are: (a) private cloud, (b) community cloud, (c) public cloud, and (d) hybrid cloud. Despite the early success and popularity of the cloud platform paradigm, as well as the widespread availability of providers and tools, this new computing model comes with several challenges and risks. Developers, providers, and end users must address these challenges and risks. User privacy, data lock-in, data security, service availability, performance, energy efficiency, scalability, and programmability are issues that must be addressed. So, clouds aim to provide computing, storage, network, software, or a hybrid of those "as a service".

Among the treatments recommended for PAD, there are exercise programs that can be performed in hospitals or clinics. In a healthcare setting, the program necessitates direct supervision by health professionals, which requires frequent patient visits to hospitals, resulting in significantly high costs and patient travel. For patients in outpatient exercise, walking is suggested by health specialists while being monitored via information and communication technology. This program is a cost-effective way to supervise PAD patients' physical activities. However, evaluating patients' development only in communication may be challenging. Therefore, an evaluation method for walking measurements is required. In a study by Piteo *et al.* [8], a method was developed to automatically detect the onset of lameness based on the analysis of data acquired from patients' smartphones as it is shown in Fig. 1.

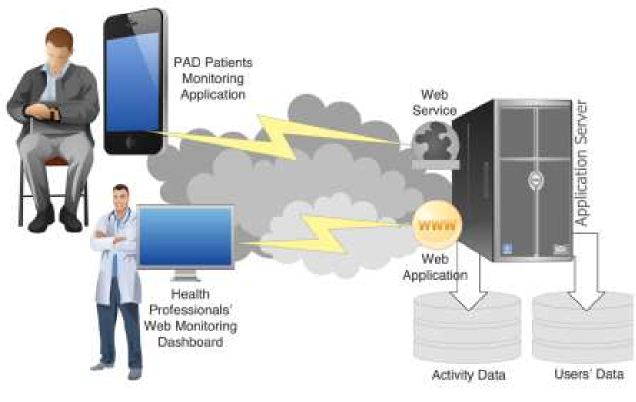


Fig. 1. System architecture of WalkingPAD [8].

They used a smartphone with an IMU 9-axis STMicroelectronics LSM6DSL to obtain an optimal motion experience for the consumer. The data were collected in three planes of motion: (X-axis), (Y-axis), and (Z-axis). The data for this exploratory study were obtained from 40 individuals with PAD. It was found that by using inertial sensors and specific gait metrics, the onset of claudication can be recognized. The model's accuracy for the dataset utilized in this study was 95.58 %. However, this work has certain limitations, such as the small data set used, as well as the fact that it was conducted while walking in a controlled environment with a clear path and no slope, both of which impacted the model's robustness.

By developing a cloud platforms to treat diseases, Ivanović *et al.* [9] presented an innovative solution for constructing and cloudifying the *in-silico* research platform SilicoFCM (Fig. 2), an innovative clinical trials' solution for the design and functional optimization of whole heart performance and monitoring effectiveness of pharmacological treatment, with the aim to reduce animal studies and human clinical trials. The primary aim of cloudification was to prove portability, increase scalability, and lower long-term infrastructure expenses.

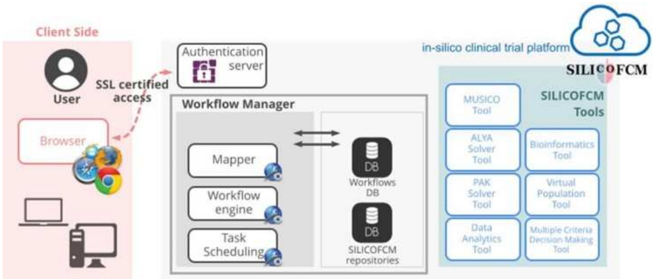


Fig. 2. Conceptual architecture of SILICOFCM [9].

As a result of connecting devices and applications used to detect and treatment for atherosclerosis treatment, a new platform called InSilc (www.insilc.eu) has been developed to create and evaluate bioresorbable vascular scaffolds (BVS). Karanasiou *et al.* [10], developed the InSilc platform, consisting of several simulators as shown in Fig. 3.

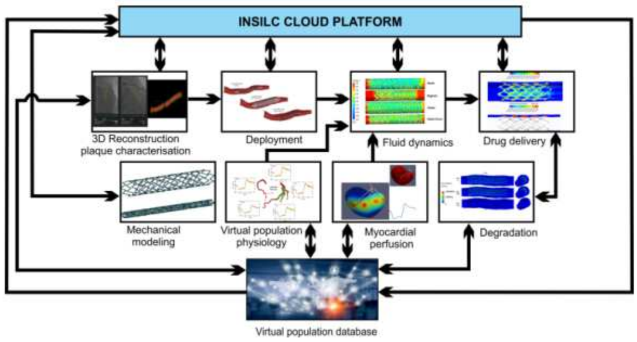


Fig. 3. InSilc cloud platform [10].

In Fig. 3, the InSilc platform is presented and its simulation tools/modules, some of which can be deemed stand-alone modules, and this is a significant advantage of the InSilc modules, as it allows the platform and modules to reach a wide variety of the market and interested stakeholders.

In the field of biology and related sciences, the DiversityNaviKey app is designed as a diagnostic tool by Triebel *et al.* [11]. It enables the diagnosis of groups of objects using queries on structured sources of descriptive data (trait data) based on combinations of optionally updated descriptor states or values that are chosen sequentially during the diagnostic procedure. DiversityNaviKey is a PWA that makes use of browser caching features like Service Worker and IndexedDB. In the study by Weber *et al.* [12], a cloud-based microbiome data analysis tool (Nephele) was created with standardized pipelines, as well as an intuitive web interface for converting raw data into biological insights. Nephele incorporates widely used microbiome analysis techniques as well as priceless reference datasets, such as the Human

Microbiome Project's cohort of healthy human individuals. The amazon web services cloud, which offers centralized and automated storage and computation power, is the foundation on which Nephele is based, facilitating thus its use by researchers and their institutions.

II. MATERIALS AND METHODS

In this section, the methodology used to implement the initial design of the DECODE cloud platform (CP) is presented. The design of the conceptual architecture is presented with all the modules and tools.The DECODE CP framework has the form of a conceptual hierarchical multilayer schema which consists of five layers. The main app of the DECODE CP is divided in two parts for front-end and back-end.

A. Conceptual architecture

The DECODE CP conceptual architecture is depicted in Fig. 4. It depicts the platform modules (e.g., user access any modules, data collection and storage module, drug coated balloon (DCB) geometries and properties storage module, image data processing and reconstruction module, modelling module, and visualization module). These modules are presented in detail with their corresponding submodules and interactions in the next section.

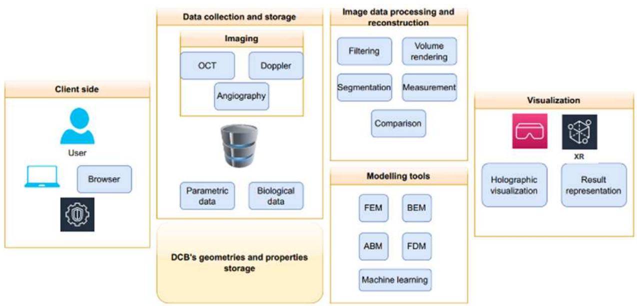


Fig. 4. Conceptual architecture of the DECODE cloud platform.

As it is shown in Fig. 4, the data providers (data uploaders) can log into the platform (through a web app) and upload data to the repository, download data from the repository, and execute converter commands on the stored data. The data processors can log into the platform (through a web app) and gain access to a variety of data analytics services (through the image data processing and reconstruction module) in order to perform: filtering, volume rendering, segmentation, measurement and comparison. Moreover, tools of computational modelling will be provided such as (i) the Finite Difference Method(FDM) is among the first techniques for resolving differential equations. When pixels are sampled in space on a regular basis, it has the benefit of creating geometries easily, directly from images, enabling the utilization of voxel centers as computational nodes directly [13], (ii) the Finite Element Method (FEM) is a numerical method that is more efficient and powerful when applied to real-world problems that include complicated geometries and boundary conditions [14], (iii) the Boundary Element Method (BEM) is a numerical method for solving linear partial differential equations in integral form for the purposes of fluid mechanics, fracture mechanics, and contact mechanics [14], (iv) Agent-based Modeling (ABM) systems are represented as groups of independent decision-making entities called agents. Each agent independently assesses its circumstances and makes decisions in accordance to a set of rules. Depending on the system they represent, agents can carry out a range of tasks. An agent might be able to learn from its experience and modify its behavior. It must have enough memory in order to learn and adapt on its own. This memory is typically represented by a dynamic agent feature. Advanced ABM occasionally integrates neural networks, evolutionary algorithms, or other learning techniques to enable realistic learning and adaptation [15], and (v) machine learning techniques can solve variable interactions and nonlinearity. Support Vector Machines (SVMs) [16] are algorithmic implementations of statistical learning theory that create consistent models from data in order to categorize newer data into defined categories. In the field of vascular medicine [17], machine learning has outperformed traditional regression models in terms of prediction accuracy. SVMs are used to categorize patients who have undergone a DES procedure into high-risk and low-risk categories based on the patient/procedural data supplied by the vascular surgeon. Additionally, users working in drug materials can directly access the DCB's geometries and properties storage module to upload and update the platform with any new material for optimization purposes. Also, in the client side, the users can access the stored 3D WebGL models to visualize the data [18].

B. DECODE cloud platform framework

The DECODE CP framework will be established as a conceptual hierarchical multilayer schema (Fig. 5). The framework consists of five layers. The first layer is the hardware layer including the cloud resources, e.g., CPUs, RAM, VMs, and additional resources. The security layer is the second layer where the OAUTH2 app programming interfaces (API) framework is incorporated along with additional mechanisms for user authentication, user access management, and secure communication [19]. The workflow layer, which is the center of the services, is located in the third layer. All of the engines that are managed by the Workflow Manager Module are included in the workflow layer [20]. These engines include the workflow engine, docker engine, and visual analytics engine. Each one is directly involved with the DECODE tools and modules lying on the fourth layer. Then, the back-end layer includes a Web App, Web Service and Database to store the different types of data (e.g. imaging, clinical data). The final layer is the front-end layer which includes the user interface and the visual analytics. As shown in Fig. 5, the platform user gets access to the graphical user interface from the front-end layer through any web browser. At this point, the user can request the DECODE CP services. The Hypertext Transfer Protocol (HTTP), which delivers and receives requests between two hosts through a secure tunnel, is used to establish a connection and carry out communication between the cloud server and the web browser. On demand, the back-end layer receives user requests and classifies them based on their relation to the platform's tools or modules. The user's requests are received by the workflow layer, which responds to them. The Workflow Manager Module which lies in the middle of the cloud layers is the core of the workflow layer. The workflows are executed through representational state transfer (REST) APIs that transfer the workflows and return the results in JSON formats [21]. The Workflow Manager Module interacts with the security layer for user authentication, user access management and communication between the cloud layers is carried out via secure tunnels. Finally, the hardware layer includes all the necessary hardware infrastructure for the establishment of the virtual machines (VMs) e.g., storage units, servers, memory units, and CPUs.

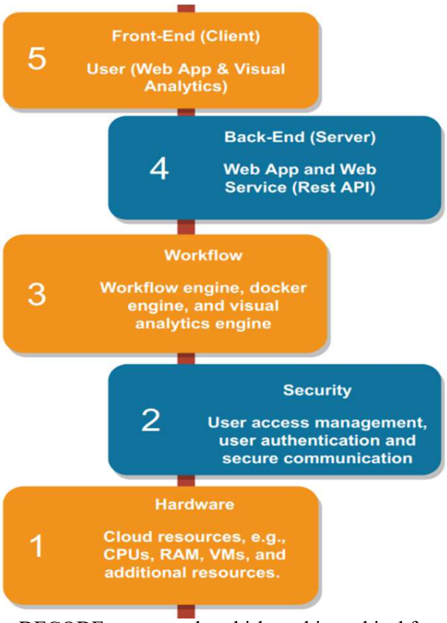


Fig. 5. The DECODE conceptual multi-layer hierarchical framework.

C. *Front-end and Back-end architectures*

Front-end app is a PWA that makes use of browser caching features like Service Worker and IndexedDB (Fig. 6). The primary tasks are thus accessible in offline mode as well. The current configuration makes use of the data pipelines, Cache Database, cloud modules and tools Descriptions as the backend for data administration [22].

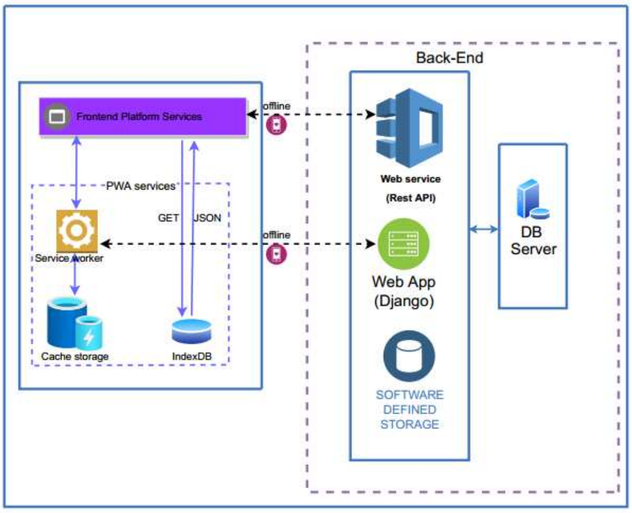


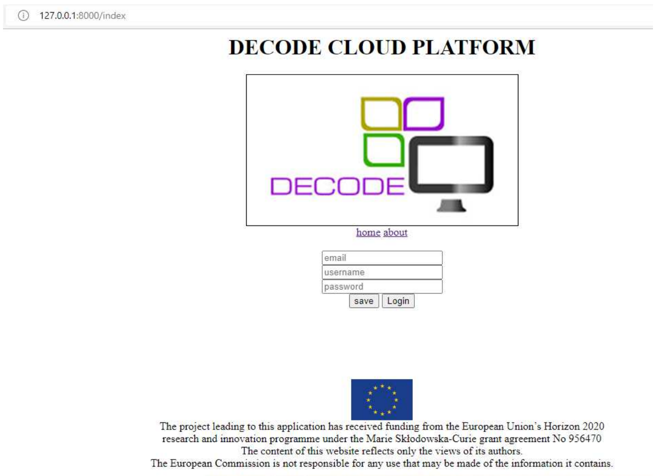
Fig. 6. Front-end with PWA and back-end with web app and service.

As far as front-end is concerned (Fig.6), the web app is a single-page application (SPA). SPAs are web apps that just need one pre-generated HTML page to load. The entire presentation logic for dynamically altering the page in the browser is contained in this page. A request is made to the server web app the first time the client app is opened in a browser. The server responds to all app data, such as HTML, Cascading Style Sheets (CSS), and JavaScript (JS). The service worker is installed at the first time the client app is called. It checks and caches updates to allow offline use of the app. The web service is used to retrieve content data, which is then saved to the browser's IndexedDB.

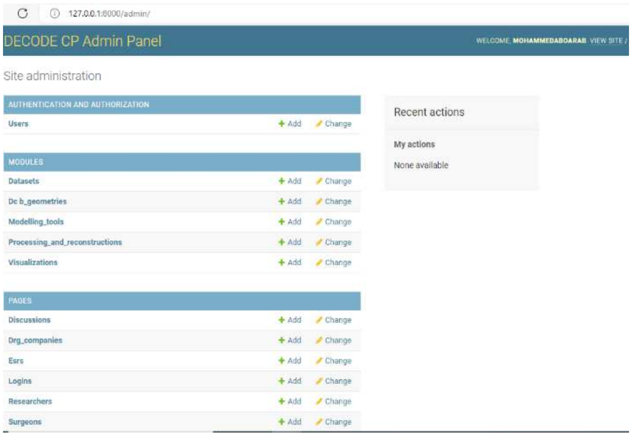
Concerning back-end, a web-based endpoint from which the SPA can retrieve data is required. A REST web service provides this endpoint. It refers to a stateless service that sends data to the requesting web app using the HTTP protocol. Uniform Resource locators (URLs) are used to identify the resources that are accessible through web services, and JSON is used to deliver the data. The web app and database are linked by the implemented web service which converts the data into the appropriate JSON format. PostgreSQL is installed as the database backend. This installation is part of the data transfer to the PostgreSQL database. The web app uses the web service as defined interface to access the PostgreSQL cache database and retrieve the desired data [23]. The results are presented in the next section, based on the conceptual architecture of the DECODE CP.

III. RESULTS

This section presents the results obtained from the proposed architecture of the DECODE CP which is implemented using the visual studio code program and Django framework programming version 4.0.5, as well as HTML, CSS and JS programming languages. The design implementation is divided into two parts: (i) Front-end app considering a login page for each user of the DECODE CP as shown in Fig. 7(a), (ii) Back-end app as it is shown in Fig. 7(b) which contains the modules app that consists of Datasets, Processing and reconstruction, Modelling tools, Visualizations, DCB geometries, pages app for the main users of the platform (e.g., Researchers, Surgeons, Drug companies, students) and login page for any user that want to use the DECODE CP. Also, it contains a discussion page for any notes and meetings between the platform users. When the users access the platform by using email, username and the password, the data of users will be saved in the login page of the database (Fig. 7). The module's apps can be accessed by the users. The datasets module, which considers a container of all datasets uploaded by the users, is shown in Fig. 8.



a



b

Fig. 7. The DECODE cloud platform: (a) the Front-end login page, and (b) the Back-end app.

In the client app, the users can upload and download any type of dataset from the datasets module and mention the name of the dataset, retrieve a description about it, the number of images and the type of dataset images (e.g., CT, MRI, OCT and CDU).

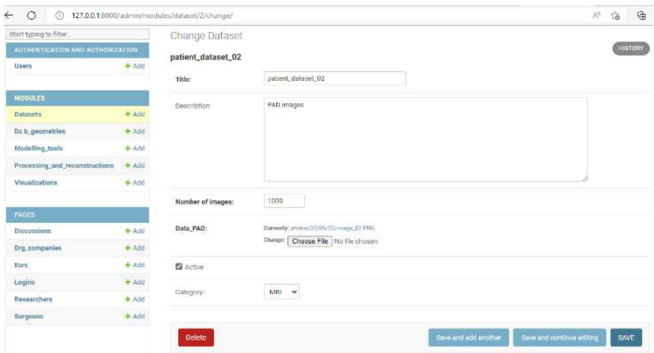


Fig. 8.The dataset module of the DECODE cloud platform.

After uploading and saving the dataset images as it is shown in Fig. 8, the user can utilize any other module. Users can access the visualization module that contains a holographic representation, augmented and virtual reality. The dataset images can be represented on the client web browser.

IV. DISCUSSION

The initial design of the DECODE CP was presented based on questionnaires collected from the main users of the platform, which are researchers, surgeons, students and medical device / pharma companies. The results were presented considering two apps, one for the front-end app and the other for the back-end app. In the front-end app, the user have the authority to deal with the database of datasets, and then to perform processing steps on the image dataset and visualize it. In the back-end app, a database is designed for the main users of the DECODE CP. This app is divided into two apps, one as a database for the dataset images collection and the other for the modules and tools for the platform. The front-end app will be based on the principle of a PWA which allows users to deal with the platform, whether online or offline, through the service worker. As for the back-end app, it will be based on the REST API. By comparing the initial design of the DECODE CP with [8, 9, 10], we found that the initial design of the DECODE platform contained an app or pages for researchers, vascular surgeons, and various users to facilitate communication between different groups that help in the development and treatment of PAD. This design also helped in saving time and cost that the patient needed to visit the treatment centers. It also helped researchers in providing the various datasets they needed in research studies.

The future work on this design will be related to authenticating users using JWT authentication to protect the DECODE CP from any attackers. The PWA feature will be implemented and, we will work on communicating with the different medical platforms and applications so that it will finally end to be an integrated work environment.

V. CONCLUSIONS

This study presented an initial design of the DECODE CP which contains a set of modules (e.g., data collection and storage module, DCB's geometries and properties storage module, image data processing and reconstruction module, modelling tools, visualization module). The design implementation is divided into two parts: (i) the front-end app which considers a login page for each user of the DECODE CP, and (ii) the back-end app which contains the modules and pages apps for the main users of the platform. It contains also a discussion page for notes and meetings between the platform users.

REFERENCES

[1] Monika Colombo, Anna Corti, Scott Berceli, Francesco Migliavacca, Sean McGinty, and Claudio Chiastra. PLoS one, "3D modelling of drug-coated balloons for the treatment of calcified superficial femoral arteries," vol. 16, no. 10, p. e0256783, 2021.

[2] Amir Bahmani, Arash Alavi, Thore Buergel, Sushil Upadhyayula, Qiwen Wang, Srinath Krishna Ananthakrishnan, Amir Alavi, Diego Celis, Dan Gillespie, Gregory Young, Ziye Xing, Minh Hoang Huynh Nguyen, Audrey Haque, Ankit Mathur, Josh Payne, Ghazal Mazaheri, Jason Kenichi Li, Pramod Kotipalli, Lisa Liao, Rajat Bhasin, Kexin Cha, Benjamin Rolnik, Alessandra Celli, Orit Dagan-Rosenfeld, and Michael P. Snyder. Nature communications, "A scalable, secure, and interoperable platform for deep data-driven health management," vol. 12, no. 1, pp. 1-11, 2021.

[3] Robin Nunkesser. "Beyond web/native/hybrid: a new taxonomy for mobile app development," in 2018 IEEE/ACM 5th International Conference on Mobile Software Engineering and Systems (MOBILESoft), 2018, pp. 214-218: IEEE.

[4] A. Francis Saviour Devaraj, Mohamed Elhoseny, S. Dhanasekaran, E. Laxmi Lydia, and K. Shankar. Journal of Parallel Distributed Computing, "Hybridization of firefly and improved multi-objective particle swarm optimization algorithm for energy efficient load balancing in cloud computing environments," vol. 142, pp. 36-45, 2020.

[5] Dean Hume. Progressive web apps. Simon and Schuster, 2017.

[6] Sayali Sunil Tandel, and Abhishek Jamadar. International Journal of Innovative Research in Science, Engineering and Technology, "Impact of progressive web apps on web app development,"

vol. 7, no. 9, pp. 9439-9444, 2018.
[7] Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger and Dawn Leaf. NIST special publication, "NIST cloud computing reference architecture," vol. 500, no. 2011, pp. 1-28, 2011.
[8] Bruno Miguel Ribeiro Pinto. "WalkingPAD-Smart sensing," 2021.
[9] Miloš Ivanović, Miloš Ivanović, Nikolaos Tachos, George Gois, Nenad Filipović, and Dimitrios I. Fotiadis. "In-silico Research Platform in the Cloud-Performance and Scalability Analysis," in 2021 IEEE 21st International Conference on Bioinformatics and Bioengineering (BIBE), 2021, pp. 1-6: IEEE.
[10] Georgia S. Karanasiou, Panagiota I. Tsompou, Nikolaos Tachos, Gianna E. Karanasiou, Antonis Sakellarios, Savvas Kyriakidis, Luca Antonini, Giancarlo Pennati, Lorenza Petrini, Frank Gijsen, Ted Vaughan, Christos Katsouras, Lampros Michalis, Dimitrios I. Fotiadis. An in silico trials platform for the evaluation of stent design effect in post-implantation outcomes. In 2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) (pp. 4970-4973). IEEE.
[11] Triebel, Dagmar, Ariane Grunz, Stefan Seifert, Anton Link, and Gerhard Rambold. INFORMATIK, "DiversityNaviKey, a Progressive Web Application for interactive diagnosis and identification," 2021.
[12] Nick Weber, David Liou, Jennifer Dommer, Philip MacMenamin, Mariam Quinones, Ian Misner, Andrew J. Oler, Joe Wan, Lewis Kim, Meghan Coakley McCarthy, Samuel Ezeji, Karlynn Noble and Darrell E. Hurt.. Bioinformatics, "Nephele: a cloud platform for simplified, standardized and reproducible microbiome data analysis," vol. 34, no. 8, pp. 1411-1413, 2018.
[13] Mikhail Shashkov, and Stanly Steinberg. Conservative finite-difference methods on general grids. CRC press, 2018.
[14] Pepper, Darrell W., Alain J. Kassab, and Eduardo A. Divo. An introduction to finite element, boundary element, and meshless methods with applications to heat transfer and fluid flow. American Society of Mechanical Engineers, vol. 334, 2014.
[15] Macal, Charles, and Michael North. "Introductory tutorial: Agent-based modeling and simulation." In Proceedings of the winter simulation conference, pp. 6-20. IEEE, 2014.
[16] N. Komal Kumar, G.Sarika Sindhu, D.Krishna Prashanthi,and A.Shaeen Sulthana. "Analysis and prediction of cardio vascular disease using machine learning classifiers." In 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 15-21. IEEE, 2020.
[17] Doyeon Hwang, Joo Myung Lee, Seokhun Yang, Mineok Chang, Jinlong Zhang, Ki Hong Choi, Chee Hae Kim. Cardiovascular Interventions, "Role of post-stent physiological assessment in a risk prediction model after coronary stent implantation," vol. 13, no. 14, pp. 1639-1650, 2020.
[18] Zhang, Qi. "Medical Data and Mathematically Modeled Implicit Surface Real-Rime Visualization in Web Browsers," International Journal of Image and Graphics Vol. 22, No. 04, 2250027, 2022.
[19] Parikshit N. Mahalle, Shashikant S. Bhong, and Gitanjali R. Shinde. Authorization and Access Control: Foundations, Frameworks, and Applications. CRC Press, 2022.
[20] Avinash Kaur, Pooja Gupta, Manpreet Singh, and Anand Nayyar. Scalable Computing: Practice and Experience, "Data placement in era of cloud computing: a survey, taxonomy and open research issues," vol. 20, no. 2, pp. 377-398, 2019.
[21] Lange, Kenneth. "The little book on REST services." Dostupno na: https://www. kennethlange. com/books/The-Little-Book-on-REST-Services. pdf (2016).
[22] Tal Ater. Building progressive web apps: bringing the power of native to the browser. " O'Reilly Media, Inc.", 2017.
[23] Nuno Pombo, and Caio Martins. "Test Driven Development in Action: Case Study of a Cross-Platform Web Application," in IEEE EUROCON 2021-19th International Conference on Smart Technologies, 2021, pp. 352-356: IEEE.