WEB APPLICATION FOR PERSONAL DIGITAL HEALTH RECORDS

Alexander BRECKO*, Ladislav POMŠÁR*, Iveta ZOLOTOVÁ*
*Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 042 00 Košice, tel. 055/602 5115, E-mail: alexander.brecko@tuke.sk

ABSTRACT

This article explores the impact and benefits of digitizing health records, focusing on the care of patients lying at home in the UK. Although electronic health records (EHRs) have existed for some time, their widespread adoption has escalated in the last decade, revolutionizing patient care. Digitization of health records increases accessibility, improves communication between healthcare professionals, and offers better monitoring and transparency in patient care. Through a case study of implementing a digital solution for patients and doctors, this article aims to show how such technological advances can significantly improve patient care, reduce costs, and provide peace of mind for patients' families.

Keywords: *web application, digitalization of personal health records, React application, Express.js*

1. INTRODUCTION

Understanding how electronic medical records could be utilized to facilitate and improve patient care has long been a challenge for doctors and researchers. While hospitals started implementing information technology in the 1960s, electronic health records have only gained broad traction in the last ten years. Healthcare professionals can benefit significantly from the digitization of patient records. These opportunities include instant access to and processing patient data through search capabilities, automated medication reminders, improved information sharing within the medical team, and increased transparency through comprehensive and readable patient condition documentation. There are now more options for treatment and care than ever, leading to more patients needing long-term healthcare. For this reason, the digitization of health records is becoming an increasingly important part of modern healthcare [1].

Digitization of medical records is the process of moving from paper copies of medical records to electronic records that are stored and maintained in digital form. This process has a major impact on the delivery of health care and improves the accessibility and management of patient data. Through digitization, healthcare providers can gain faster and more efficient access to patient information, thus providing faster and more effective care. In addition, digitization of health records reduces the cost of paper documentation and increases the security of patient data. An electronic health record is a digitized medical record. Getting value from this technology requires a wide range of functions that collect, manage, and share digital health information. This information can then be used to support medical decision-making and clinical operations. Ideally, information gathering begins before a patient visit by obtaining records from other providers or previous patient encounters. This and other information are then updated at the beginning of the patient's interaction with the physician or nursing staff; additional data-such as lab results, images, and progress notes-are added as the encounter progresses. Ideally, this data could be portable to be shared with other providers or accessed through patient portals [2].

The digitization of health records not only represents a revolution in the efficiency and accessibility of healthcare but also brings tangible benefits to patients, healthcare professionals, and, not least, patients' family members. This article will focus on a specific case of the application of digitization: the care of home lying-in patients in the UK. This article looks closely at how digital technology is used to help patients who are confined to their homes in the UK. We chose to focus on the UK for this study because of a special chance to work together with healthcare providers there. Our team, which is based in Slovakia, was given access to medical records and documents from UK guardians. This help was crucial for us to develop a web application that meets the needs of the UK healthcare system. The collaboration went beyond just making a web application. It included working with technology in other ways too, like collecting data from sensors, processing this data, and then analyzing it. Getting to use real healthcare data from the UK and having UK guardians willing to work with us played a big part in why we decided to concentrate our research in the UK. In recent years, we have seen how digitization has enabled better patient health monitoring, simplified communication between healthcare professionals, and provided patients' families with invaluable peace of mind and control. Using the example of implementing our solution for home-lying patients in the UK, we show how technology can be used effectively to improve care and support those who rely on it.

2. RELATED WORK

Health record digitization applications are software tools that enable the digitization and storage of patient health records. These applications usually contain information on a patient's health status, medications and prescriptions, tests and examinations, diagnoses, and surgical procedures. Some apps also enable doctor-patient communication, online appointment scheduling and management, and real-time tracking of health data and outcomes. These apps improve patient care treatment efficiency and increase patient satisfaction. The following overview highlights specific smart technologies that have found their application in practice. The common features of these applications are that they are easy to use, provide access to the most effective treatments available, ensure the highest quality of care

at affordable prices, and offer access to a wide range of services (e.g., access to consultants, diagnostic tests, therapies, and surgical procedures. There are many applications for digitizing health records. Some of the most popular are:

- **Epic Systems** is one of the largest and most widely used digital healthcare platforms in the world. This application is designed for healthcare facilities and physicians. It allows you to efficiently manage patient health records, share data and improve the overall delivery of healthcare [3].
- **Oracle Cerner Corporation** is another major digital health platform that helps physicians and healthcare providers manage patient records and coordinate care. Cerner offers a variety of tools that allow you to monitor a patient's health status, manage medications, and perform a variety of medical procedures [4].
- **Allscripts** is a popular digital health platform that allows you to manage patient health records, share data between healthcare facilities, and improve the quality of care you provide. Allscripts also offers mobile apps that allow physicians to access important patient information anytime, anywhere [5].
- **eClinicalWorks** is a cloud-based platform for digital healthcare. It specializes in healthcare providers, small and medium-sized healthcare facilities. This application allows healthcare providers to efficiently manage patient health records, coordinate care, and increase productivity [6].
- **NextGen Healthcare** is a digital healthcare platform that helps manage patient health records, increase the efficiency of healthcare delivery and improve the overall quality of care. This application is designed for healthcare facilities, clinical laboratories, physicians and enables efficient coordination of patient care [7].

Despite the progress of large companies in digitizing health records, there are various implementations and applications in the form of web applications that simplify the digitization and analysis of health records. For example, the article [8] describes the technical development and deployment of the first patient-specific genomic test report (PGR) within the electronic health record (EHR) ecosystem using a locally developed standards-based web application interface. The goal was to develop a tool that allowed the creation of patient-specific PGRs and accompanying reports for the provider. These were converted into a format that allowed them to be presented in the patient portal or EHR using an existing interface, allowing patients, caregivers, and providers to access individual reports intended for the intended end user.

Also, in the article [9], the authors try to link the Internet of Things (IoT) and EHR. They propose a high-capacity, secure, and computationally efficient technique for hiding electronic health records (EHRs) in medical images in an IoT-driven healthcare system. The scheme uses the pixel repetition method (PRM) and modular arithmetic. The PRM was used to magnify the input medical image to create a cover image, and modular arithmetic was used to embed the secret EHR into the magnified images. The proposed scheme was tested in detail for various commonly used medical/test images and a group of randomly selected images from the Uncompressed Color Image Database (UCID) repository. Experimental investigations showed that the proposed scheme is capable of providing secure and high embedding capacity while maintaining reasonable imperceptibility, in addition to reversibility. Further, the use of PRM for generating the cover image is computationally highly efficient compared to the state-of-the-art in this area. Hence, it is most suitable for electronic health record (EHR) exchange in an IoT-based healthcare system for smart city applications.

Given that this is medicine and healthcare, where data are very sensitive, several publications deal directly with the security of these data and how to share these data in the most secure way possible. Within this, we can encounter various security and encryption mechanisms. There are a number of articles and publications [10–13] that are devoted to security. Proposed systems that primarily deal with health records and data often use blockchain to secure their data at this time. Blockchain enables the creation of immutable patient logs for secure storage and quick access to medical records, updating medical records, managing health information between different providers, and peer-to-peer blockchain browsing contracts. Participating in this clinical data system provides patients with a complete distributed ledger record of all records of all events and seamless access to their healthcare provider's electronic health records.

3. ARCHITECTURE DESIGN

The objective of this paper is to design and implement a web-based application that facilitates doctors in maintaining and accessing patient records. This digital platform aims to streamline the process of updating individual patient profiles with pertinent health information while simultaneously providing a user-friendly interface for physicians to review these records as needed.

The paper will describe the development of both frontend and backend technologies. The frontend will focus on creating an intuitive and responsive user experience, ensuring the application is accessible and easy to navigate for the doctors. Concurrently, the backend development will prioritize creating an infrastructure capable of managing sensitive patient data with utmost confidentiality and integrity.

Integrating these technologies will result in a comprehensive web application that not only improves the efficiency of patient record management but also adheres to the high standards of healthcare information systems.

3.1. Frontend

Frontend [14] is the public part of web applications that allow users to interact and communicate directly with the application. The frontend includes displaying the func-

tional tasks and user interface that are performed on the client side, as well as processing requests from users. On any page, you see buttons, texts, animations, and other components - all of this is implemented using the frontend. The frontend is responsible for ensuring that the web application displays correctly on different devices and screens, such as computers, mobiles, and tablets, to provide users with an optimal experience on different platforms. In addition, the frontend can contain code that allows various client-side functions, such as logic calculation, data validation, and state management, allowing for faster responses to user actions without having to send data to the server and wait for a response. Frontend is also closely related to UX (user experience) and UI (user interface) design, which ensures that the web application is user-friendly, easy to use, and provides a positive experience for users. This is an important factor that affects the overall success and user satisfaction with the product [15].

Several web frameworks are used to simplify the work with the frontend and creation of web applications. Frontend frameworks are very useful on the presentation side of the web and create a good foundation from which to build pages. These frameworks combine HTML, CSS, and JavaScript to effectively create conventions for different elements [16]. Among the most popular frameworks for creating web applications, we can include:

- **React.js** is a popular framework developed by Facebook. This framework focuses on the creation of user interfaces. React uses JSX. JSX is an extension of JavaScript syntax that allows programmers to create user interfaces using HTML-like code [17].
- **Angular** is an open-source framework developed by Google. Angular focuses on building dynamic web applications and websites. Angular uses TypeScript. TypeScript is an extension of the JavaScript language that allows programmers to write code with greater clarity and security [18].
- **Vue.js** is a progressive JavaScript framework that focuses on creating user interfaces. Vue.js uses unidirectional data bindings, which allow programmers to control the state of the application. Vue.js also allows programmers to use composite templates that make it easier to create user interfaces [19].

All the mentioned frameworks have their strengths and weaknesses. Except for a few differences, they solve similar problems. In this work, any framework could be chosen to create an application for digitizing medical records. After examining the available options, analyzing statistics, and based on my own experience, it was decided to use React.js as a framework for frontend implementation.

3.2. Backend

Backend [20] refers to a piece of software hidden from the end user. While the frontend focuses on the user interface and what the user sees and interacts with, the backend deals with data processing, databases, security, and server logic [21]. The backend handles requests that come from the frontend. With the backend, several options and frameworks allow you to create a backend:

- **Express.js** or simply Express, is a backend web application framework for building RESTful APIs with Node.js, released as free and open source software under the MIT license. It is intended for creating web applications and APIs [22].
- **Flask** is a micro web framework written in Python. It is classified as a microframework because it does not require special tools or libraries. It has no database abstraction, form validation, or any other components where already existing third-party libraries provide common functionality [23].
- **Django** is a high-level web application framework written in Python. It was designed to facilitate the development of complex web applications with an emphasis on speed, security and code maintainability [24].

Given that we chose the very popular React for the frontend, Express.js, which is directly intended as a backend framework for Node.js and React, will offer us the greatest advantage and compatibility for the backend.

3.3. Communication

Communication between frontend and backend takes place via protocols such as HTTP (Hypertext transfer protocol) or HTTPS (Hypertext Transfer Protocol Secure). HTTP is an application layer protocol designed to transfer information between network devices and runs on top of other layers of the network protocol stack. The frontend sends a request to the server via these protocols, and the backend responds with the necessary data.

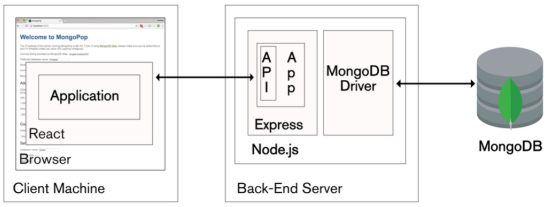

Fig. 1 Architecture of communication

Our communication architecture is shown in Fig. 1. As mentioned above, a web application programmed using React will run on the user's side. This web application will communicate with our backend. This backend will also communicate with our MongoDB database.

3.4. Safety

It is possible to log in to the system using a login name and password. This system architecture delineates specific roles for its users, with a particular emphasis on two primary roles: patients and doctors. For patients, the system grants exclusive access to their individual health records

and personal messages that are archived within the system. Conversely, the role of the doctor or caregiver is endowed with a broader scope of access. Specifically, doctors possess the capability to access the records of all patients under their care, including those they have personally added or managed within the system. This delineation of access rights is integral to maintaining the confidentiality and integrity of patient information while ensuring efficient healthcare management by medical professionals.

4. ANALYSIS OF PAPER FORMS

Before the design and development of the application, it was necessary to examine the paper forms which are normally used in home care. Analyzing paper forms before their digitization is important because it allows one to get an overview of what information is to be obtained and what data is to be displayed. In this way, we designed and optimized the digitization process in advance so that it is efficient and minimizes the number of errors during data collection and storage. The analysis of paper forms also makes it possible to identify potential problems in the digitization process and to find solutions to prevent or minimize these problems.

The Fig. 2 shows the elevation chart form, which is used for thorough monitoring and control of the patient's health, emphasizing the angle of inclination of the head and the duration of this position. This form contains mostly textual information that is systematically organized into 6 columns. The individual columns contain important data, such as the patient's name, the date and time of the recording, the angle of the patient's head, the duration of the given position, and the caregiver's signature.

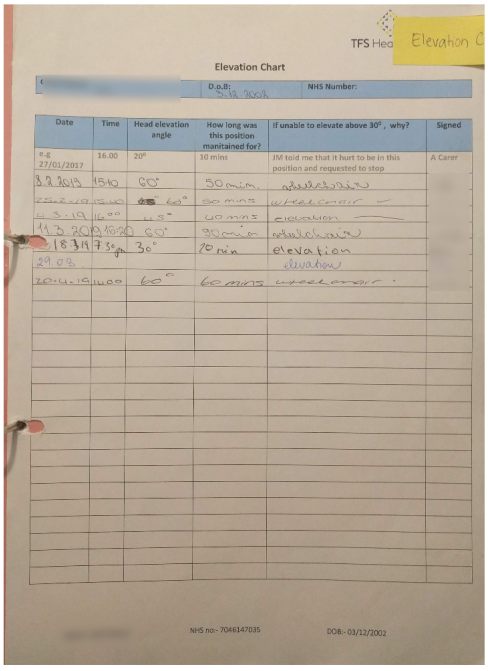

Fig. 2 Paper version - elevation chart

The child daily care report - another form that is used to record information about the child's care during the day. It contains various text fields where data on the child's speech status, verbal and non-verbal communication, concentration, as well as feedback from family and friends are recorded. In addition, this form also records information about medication administration and other important data related to the care of the child. Like other forms, the Child Daily Care Report contains fields for recording the patient's name, date, time, and caregiver's signature. These data are used to identify and monitor the care of the child and provide a record of the time when the care was provided.

The pressure relief positioning chart is a form used to minimize pressure and friction on certain parts of the body that can potentially lead to skin ulcers. This form contains various text and numeric fields, divided into 6 columns, where important information such as the patient's name, date, time of recording, skin condition, recorded position, and pressure relief measures taken are recorded. In addition, each record is signed by the caregiver, which ensures accountability for the measures taken. A paper version of this form is shown in Fig. 3.

The pediatric tracheostomy safety checklist is a form that contains a list of measures that the caregiver must check and record on the form to verify that all necessary tracheostomy procedures have been performed in pediatric patients. This form contains fields for recording the patient's name, date, time and caregiver's signature. The form is specific in that it is divided into a table according to the days of the week, and each day is further divided into day and night shifts. The form does not use a normal text format, instead the caregiver records data in the form of *yes* or *no* (boolean format), which distinguishes it from normal text forms. Fig. 4 shows the paper version of this form.

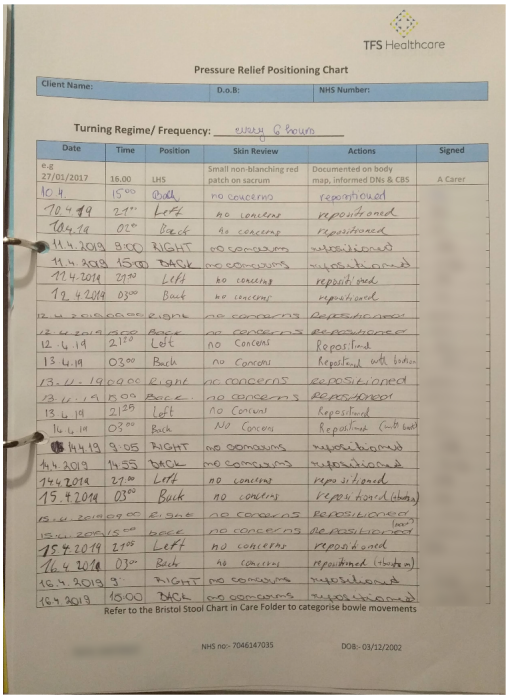

Fig. 3 Paper version - pressure relief positioning chart

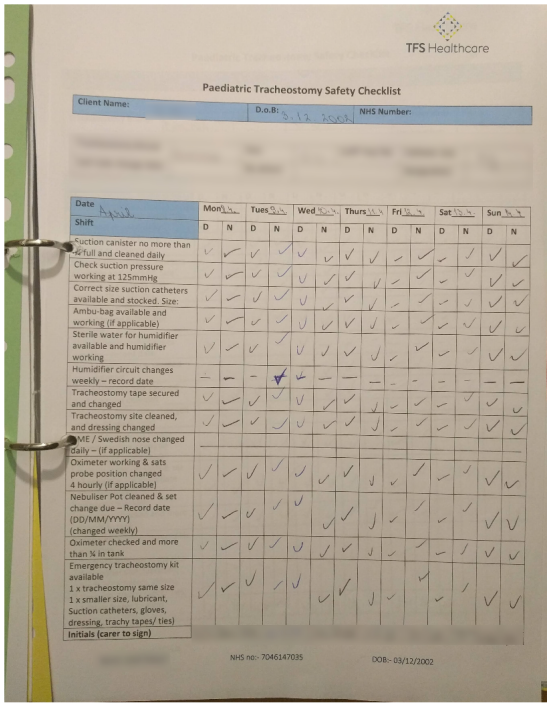

Fig. 4 Paper version - paediatric tracheostomy safety checklist

The body map form is a special tool used by healthcare professionals to record important information about a patient's health. This form contains an image with a drawing of the patient's body, which includes both front and back views. On the image, the healthcare professional can mark problem areas such as wounds, burns or bedsores. In addition, the form also contains information such as the patient's name, date and time, notes and the caregiver's signature. A paper version of this form is shown in Fig. 5.

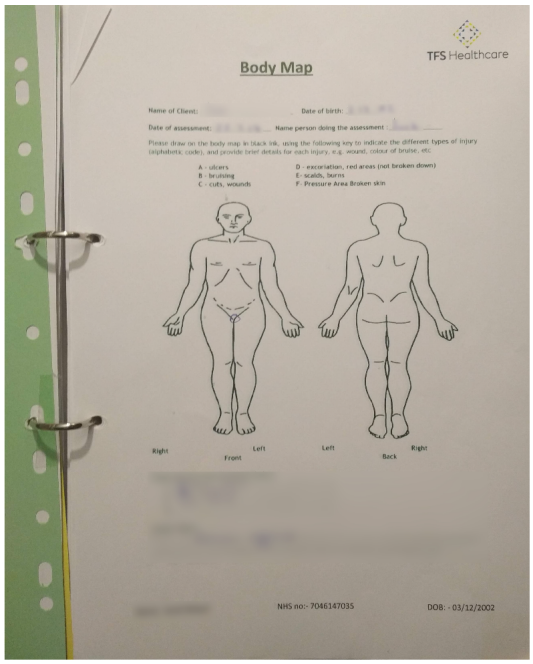

Fig. 5 Paper version - body map

The process of digitizing paper forms consisted of several steps. The first step was to obtain the paper forms and identify their type and purpose. The purpose of these health forms was to record and store information about patients, their health status and treatment, so that it would be available to medical personnel and health care providers.

The next step was the identification of patterns and common elements on the forms, which were subsequently used for the design of the digital form. On all forms, we found that certain data were repeated, such as date, time, patient's name, and the name or signature of the caregiver, which must be written for each entry. These are common elements that appear on all forms. Based on this information, we created a conceptual model of the form and its functionality.

5. WEB APPLICATION DESIGN

When implementing web applications, it was first necessary to draw how the individual subpages would look and what they would contain. Our goal was to design a web application that would completely replace the paper form and provide the user with the same information. All the digital forms mentioned above have a very similar functionality, which consists of adding a new medical record, editing or deleting existing records and displaying all records in the form of a table. Entering values when adding a new record will take place in a separate window. Additional buttons will be programmed to modify, edit and delete some records. For each paper form, a separate subpage was created, which has a certain functionality.

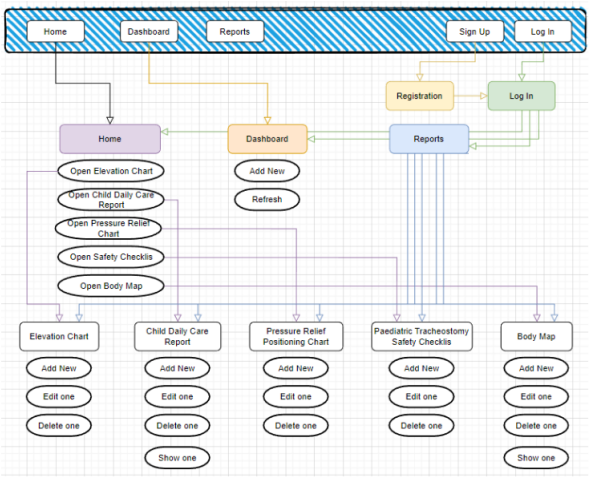

Fig. 6 Navigation scheme on the web application

The navigation architecture of a website refers to the way the navigation is organized and structured. This determines how the user can navigate between pages or sections of a website. It is a plan, scheme or model that determines how different parts of a website are interconnected and how the user is allowed to access individual pieces of content. The importance of a website's navigational architecture is that it allows users to easily and quickly navigate

the site, find the information they want, and achieve their goals. A well-designed navigation architecture increases the usability of a website, improves the user experience, and contributes to an efficient and satisfying user experience. The web application navigation scheme proposed by us is shown in Fig. 6

For a better understanding of the architecture of our web application for digitizing health records, it is necessary to describe the individual pages and their available functions for users in more detail.

Homepage: This page contains an overview of all available forms of medical records. Here, users can see a short information about each form and have the option to click on the corresponding button to open a specific form.

Form page: Here the user can see the complete form with all the added entries. Various functions are available for the user on the form page, such as adding a new form, editing the content of the form, deleting the form or opening a detailed version of the form. Here, the user can perform various actions with the form according to his needs. When adding or editing a record, a simple form opens where the user fills in the necessary data. To delete a record, just press the corresponding 'Delete' button next to the record. If the user enters incorrect values in the form, the system will notify him of this error.

Registration: When registering, the user registers using a simple form, where he must fill in a username, email address and password (which must be entered twice). If the user selects an existing email address, the system will notify him that there is already a user with such an email address.

Login: This page is intended for the login of users who are already registered in the system. The form consists of two parts - user e-mail address and password. In the case of an unsuccessful login attempt due to an incorrect e-mail address or password, a warning will be displayed on the screen.

Dashboard: Here the user can see graphs with patient information over time. The user has the option to add additional data using the form, which will be displayed on the chart at the current time point. If the user enters incorrect values in the form, the system will notify him of this error.

6. IMPLEMENTATION OF WEB APPLICATION

When developing and designing the design for the healthcare application, we paid close attention to the needs and requirements of the target user group, which are healthcare professionals and patients. We designed the design to be simple, clear and intuitive, with an emphasis on usability and accessibility. We also took into account the principles of healthcare design, such as clear legibility, colour contrast and appropriate content layout. The aim of the web app is to help users solve health problems, so it is crucial that the information on the site is easily accessible and comprehensibly presented.

One of the most basic pages is the dashboard page. The dashboard is a visual overview interface that provides a quick overview of important patient information. Relevant data are displayed there, which enable a quick analysis and assessment of the health status of patients. This information is presented in a clear form, which enables simple monitoring and management of health care. The dashboard window contains a pair of graphs that provide information about the vital functions of the patient's body.

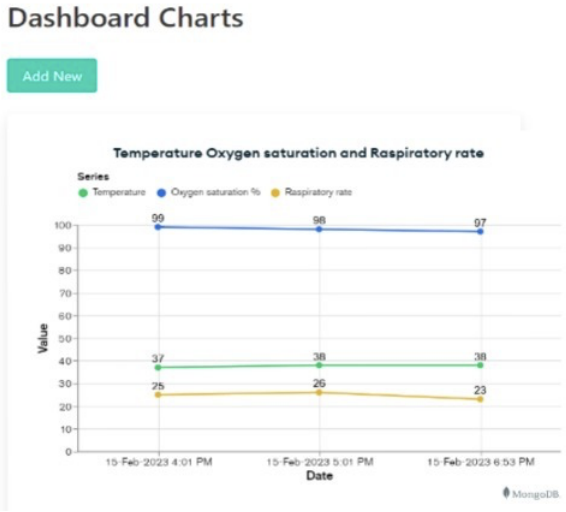

Fig. 7 Dashboard

The first graph shows the patient's temperature, oxygen saturation and breathing rate values on the Y axis, which makes it possible to monitor and evaluate these critical values. The second graph shows the values of the patient's blood pressure and heart rate on the Y axis, which provides important data on the state of the patient's cardiovascular system. On the X-axis of both graphs, the exact date and time the record was added is displayed, which allows you to follow their development over time. All data is loaded into the dashboard directly from the database, at the same time it is possible to edit and modify this dashboard as needed. The dashboard window is shown in Fig. 7

One of the most complicated subpages was the body map subpage. This is a subpage used by a health professional to record a patient's health status. The paramedic marks on the patient's silhouette the areas where the patient has problems, such as wounds, burns, and other traumatic injuries. The form also contains textual information that describes the patient's problems and metadata such as date, time of recording. It also allows you to add a new record, delete a record, view all records in the table, but unlike other forms, it includes visual information about the location of the problem on the patient's body. Therefore, it also includes a corresponding point map that allows you to view a specific record with the relevant location information. The figure silhouette consists of a background image and interactive dots that allow the user to mark the necessary areas. To create the interactive zones on the images, we used the *react-img-mapper* library, which creates a React component that accepts an image as input, a JSON map of the coordinates of specific pixels for the interactive points, and other features that provide the actual interaction. This library allowed us to create a silhouette of a character that is interactive and allows you to label individual body parts. This part can be seen in the Fig. 8.

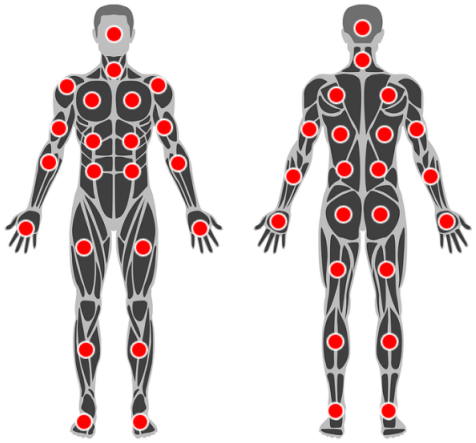

Fig. 8 Body map

Paediatric tracheostomy safety checklist protocol is a list of steps that the caregiver must check and record on the form to make it clear that all or which of the necessary procedures have been performed. In the paper version of the form, the list was divided into seven working days and each day was further divided into day and night shifts, which in practice meant that manual work was required to fill in the entries for each day. However, in the digital version, a different approach was decided upon. Instead of splitting the working days into day and night shifts, a chronological record was chosen, where the date and day of the week are added to each record. A separate column was added to represent day and night shifts, where a sun or moon icon was added to the corresponding date. The sun represents the day change and the moon represents the night change. This approach has made it easier and faster to complete the form without having to manually split the records into days and shifts. In the following Fig. 9 and Fig. 10 you can see what the digital version of the paediatric tracheostomy safety checklist protocol looks like.

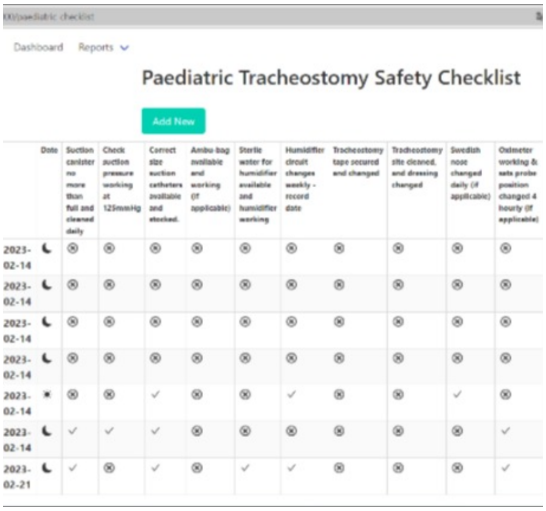

Fig. 9 Table of the digital version of the paediatric tracheostomy safety checklist protocol

In a very similar way, we also made a web application for the other forms that were analyzed in Section 4. For each single paper version of the form, a custom subpage was created where the user can enter individual values and parameters that are directly stored in the MongoDB database.

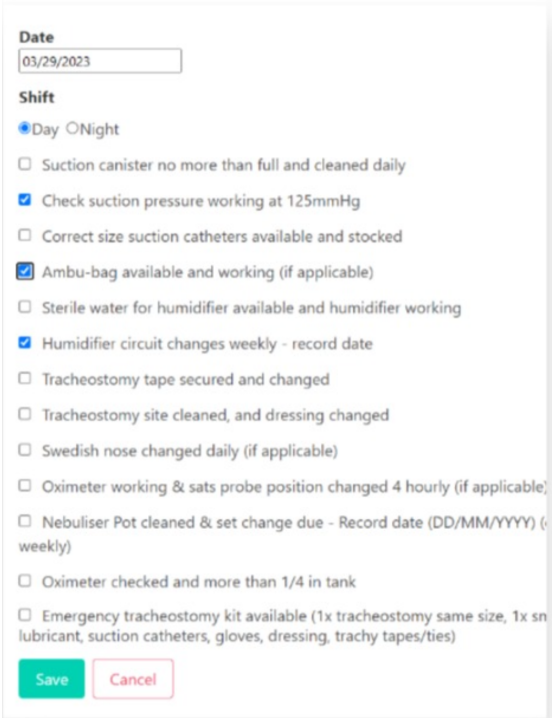

Fig. 10 Checklist for the digital version of the paediatric tracheostomy safety protocol

7. RESULTS

When developing any software, software testing is a key task. It is the method that determines whether the actual software product meets the expected requirements and ensures that the software product is free of bugs. This involves running software/system components through manual or automated tools to evaluate one or more system functionalities. The goal of software testing is to find bugs or missing requirements against the actual requirements. There is a high probability that the final code contains functionality errors and design flaws. Software testing is important because if there are any bugs in the software, they can be identified and corrected early before the software product is delivered. A properly tested software product ensures reliability, security and superior performance, leading to time savings, cost savings and end-user satisfaction.

7.1. Test scenarios

We created three test scenarios for our application, which helped us test functionality and stability. 12 respondents were contacted and asked to perform all three test scenarios we created to test the web application. The participating respondents were instructed and familiar with the basic features of the application and also knew who they were

intended for. The test scenarios and a link to our application page were in the document each respondent received. The application was developed locally and deployed on the MS Azure cloud platform for testing purposes. At the end of the testing, all participants completed a short questionnaire consisting of 8 questions. The questionnaire is an important part for getting feedback in order to fix bugs and improve the application.

Table 1 Test scenario for editing the caregiver profile

Action	Expected result
The administrator clicks the log in button.	The app will display a screen with a form to log in.
The administrator fills in the form with their login details.	The application logs the administrator in and displays a home screen.
The administrator clicks the show users button.	The application displays a page with a list of users.
The administrator selects a specific user and clicks the edit button.	The application will display the form.
The administrator fills in the form and clicks the submit button.	The application modifies the user's profile.

Table 2 Test scenario for caregiver actions

Action	Expected result
Caregiver clicks the log in button.	The app displays a screen with a login form.
Caregiver fills in the form with their login details.	The application logs the caregiver in and displays a home screen.
Caregiver clicks the show my patients button.	The application displays a page with a list of patients.
Caregiver selects a specific patient and clicks the show more button.	The application displays the patient's profile.
Caregiver clicks the daily reports button.	The app displays all the daily reports of the selected patient.
Caregiver clicks the create new report button.	The application displays a form.
Caregiver fills in the form and clicks the submit button.	The application sends the filled form to the database.

In Table 1 is a scenario designed to test the editing of the caregiver profile. The primary actor of this scenario is an administrator type user and the test scenario consists of five steps, the condition is that there must be at least one user of the caregiver type.

The second test scenario can be found in Table 2 , where we created a scenario to test the functionality for adding a daily report. The primary actor in this scenario is a user of type caregiver. The user must be registered and the caregiver position must be approved by the administrator. Our test scenario consists of seven steps.

Table 3 Test scenario for patient actions

Action	Expected result
Patient clicks the log in button.	The app displays a screen with a login form.
Patient fills in the form with their login details.	The application logs the patient in and displays a home screen.
Patient clicks the show daily reports button.	The application displays a page with a list of daily reports.
Patient clicks the show height chart button.	The application displays a page with data.
Patient clicks the show pressure relief position table button.	The application displays a page with data.

Our final test scenario is shown in Table 3 and aims to test the possibility of displaying the data to patients. The primary actor in this scenario is the patient user type. The user must be registered and the condition is that the patient has the displayed data filled in.

7.2. Evaluation of testing

This section presents an analysis of the survey outcomes, in which participants provided feedback on multiple facets of the application, including its usability, functionality, clarity, and their overall level of satisfaction or dissatisfaction. In the survey, participants had the opportunity to evaluate our application using a rating scale that ranged from 1 (indicating the highest level of satisfaction) to 5 (indicating the lowest level of satisfaction).

Upon evaluating the responses to the query regarding the appeal of the application's design, it becomes evident that the participants were more than satisfied. These results are shown in the Fig. 11 It is fair to infer that the design did not cause any distractions; rather, it was characterized by its simplicity and clarity.

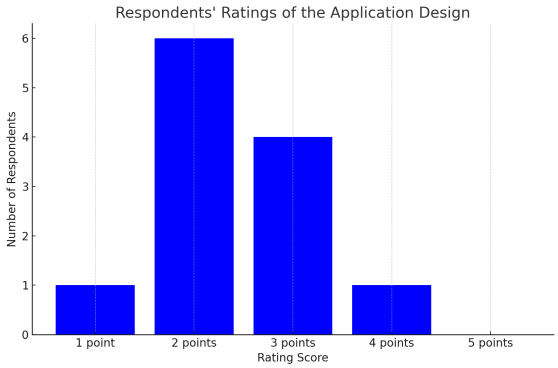

Fig. 11 Application design evaluation

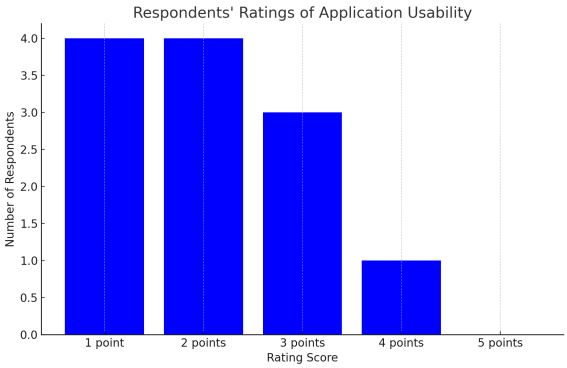

Fig. 12 Evaluation of the navigation in the application

An analysis of responses to the second question, which assessed the difficulty of using the application, is presented in Fig. 12. The majority of respondents gave positive feedback, indicating that navigating the application was not challenging.

The third question was directly aligned with our testing scenarios. We inquired if respondents encountered any issues while utilizing the various functions. The responses are depicted in Fig. 13. It is apparent from the feedback that the majority of users found functions such as login and registration, submitting daily reports, and accessing patient data to be very straightforward. However, complications were noted in the functionality related to editing the nurse's profile.

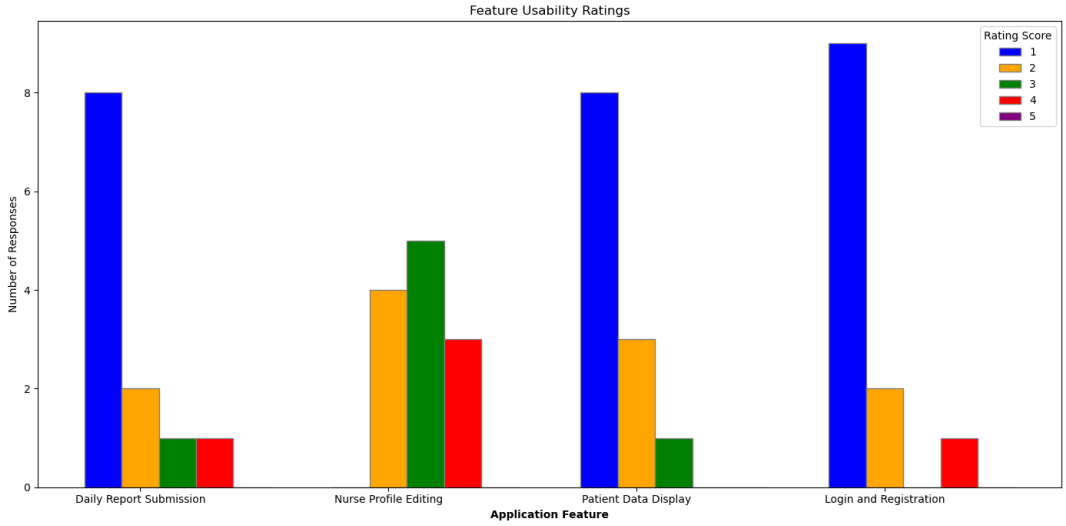

Fig. 13 Assessment of the difficulty of performing each function

The evaluation of the application's functionality and stability was addressed in the fourth question. The respondents' answers are illustrated in Fig. 14. The graph highlights that the application was stable, did not shut down unexpectedly, and operated effectively.

Other criteria were also evaluated, such as the evaluation of whether the application met expectations, the evaluation of overall satisfaction or also overall feedback, where the user had to verbally describe problems, suggestions and improvements to the application.

From the responses, we found that the functionalities in the test scenarios were accessible for the users, and no significant problems were encountered. The application was generally positively evaluated, and users had no reservations about the design and functionality.

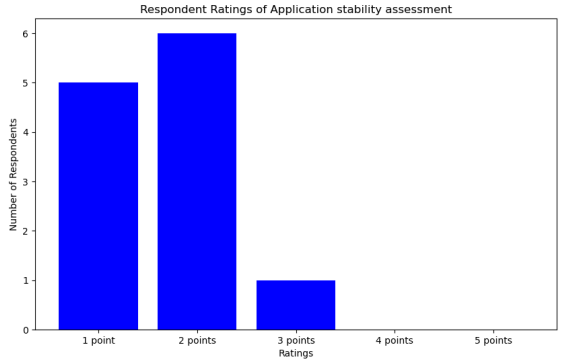

Fig. 14 Application stability assessment

8. CONCLUSIONS

We have performed a analysis of modern technologies relevant to the development of web applications. This includes an overview of the tools and frameworks for digitizing health records. During the development process, we encountered and solved various challenges, including front-end issues related to design and user interface, as well as back-end issues related to application architecture and integration between front-end and back-end components.

A thorough review of the paper forms was conducted, focusing on the details and intricacies of the information fields. After this analysis, we started designing and prototyping the web application so that its architecture and functionalities meet the required standards. This phase also included rigorous testing of the application's functions.

Subsequently, we developed an efficient system for storing information from medical records. Finally, using the React.js and Express.js frameworks, we successfully created a web application dedicated to digitizing medical records. The application uses the MongoDB database for safe and efficient data storage. This paper highlights the advancements in web application development and enables the provision of a functional and user-friendly digital solution for health record management.

ACKNOWLEDGEMENT

This publication is the result of the research project funded by VEGA grant EDEN: EDge-Enabled intelligeNt systems (VEGA/1/0480/2).

REFERENCES

[1] Pohlmann, S., Kunz, A., Ose, D., Winkler, E., Brandner, A., Poss-Doering, R., Szecsenyi, J. & Wensing, M. Digitalizing health services by implementing a personal electronic health record in Germany: qualitative analysis of fundamental prerequisites from the perspective of selected experts. *Journal Of Medical Internet Research*. **22**, e15102 (2020)

[2] Atasoy, H., Greenwood, B. & McCullough, J. The digitization of patient care: a review of the effects of electronic health records on health care quality and utilization. *Annual Review Of Public Health*. **40** pp. 487-500 (2019)

[3] Epic Systems Corporation, Epic Software, `https://www.epic.com`

[4] Oracle Cerner, Inovation for healthcare, `https://www.cerner.com`

[5] Allscripts, Veradigm formerly Allscripts, `https://veradigm.com`

[6] eClinicalWorks, Reinvent Your Practice with a Multidimensional EHR, `https://www.eclinicalworks.com`

[7] NetxGen Healthcare, EHR sustem, `https://www.nextgen.com`

[8] Williams, M., Kern, M., Lerch, V., Billet, J., Williams, J. & Moore, G. Implementation of a patient-facing genomic test report in the electronic health record using a web-application interface. *BMC Medical Informatics And Decision Making*. **18**, 1-9 (2018)

[9] Parah, S., Sheikh, J., Akhoon, J. & Loan, N. Electronic Health Record hiding in Images for smart city applications: A computationally efficient and reversible information hiding technique for secure communication. *Future Generation Computer Systems*. **108** pp. 935-949 (2020)

[10] Tanwar, S., Parekh, K. & Evans, R. Blockchain-based electronic healthcare record system for healthcare 4.0 applications. *Journal Of Information Security And Applications*. **50** pp. 102407 (2020)

[11] Chelladurai, U. & Pandian, S. A novel blockchain based electronic health record automation system for healthcare. *Journal Of Ambient Intelligence And Humanized Computing*. pp. 1-11 (2022)

[12] Kaur, J., Khan, A., Abushark, Y., Alam, M., Khan, S., Agrawal, A., Kumar, R. & Khan, R. Security risk assessment of healthcare web application through adaptive neuro-fuzzy inference system: a design perspective. *Risk Management And Healthcare Policy*. pp. 355-371 (2020)

[13] Hylock, R. & Zeng, X. A blockchain framework for patient-centered health records and exchange (HealthChain): evaluation and proof-of-concept study. *Journal Of Medical Internet Research*. **21**, e13592 (2019)

[14] Abdullah, H. & Zeki, A. Frontend and backend web technologies in social networking sites: Facebook as an example. *2014 3rd International Conference On Advanced Computer Science Applications And Technologies*. pp. 85-89 (2014)

[15] Dankanych, N., Webová aplikácia pre digitalizáciu zdravotných záznamov, `https://opac.crzp.sk/?fn=detailBiblioForm&sid=D98A4A9B442AA1687F52426ADD31`, 2023

[16] Lu, M. & Yeung, W. A framework for effective commercial web application development. *Internet Research*. **8**, 166-173 (1998)

[17] React, The library for web and native user interfaces, https://react.dev

[18] Angular, Deliver web apps with confidence, https://angular.io

[19] Vue, The Progressive JavaScript Framework, https://vuejs.org

[20] Canaday, R., Harrison, R., Ivie, E., Ryder, J. & Wehr, L. A back-end computer for data base management. *Communications Of The ACM*. **17**, 575-582 (1974)

[21] Drobňák, D., Tvorba webovej platformy Check-Up, https://opac.crzp.sk/?fn=detailBiblioForm&sid=22891320CECCAE8269CB498E7C57, 2022

[22] Express.js, Fast, unopinionated, minimalist web framework for Node.js, https://expressjs.com

[23] Flask, Flask documentation, https://flask.palletsprojects.com/en/3.0.x/

[24] Django, Django makes it easier to build better web apps more quickly and with less code, https://www.djangoproject.com

BIOGRAPHY

Alexander Brecko is a PhD student at the Department of Cybernetics and Artificial Intelligence, Technical University of Košice. He received his Bachelor's degree in 2019 and Master's degree in 2021. His Bachelor's thesis deals with edge computing. His master thesis deals with the development of a REST API for communicating multiple lab models and their synchronization. His current research deals with federated learning and edge computing in transport and health.

Ladislav Pomšár is a PhD student at the Department of Cybernetics and Artificial Intelligence, Technical University of Košice. His research is centered around deployment of AI models to the Edge, especially to the Edge AI accelerators, and medical artificial intelligence, specifically in the area of Parkinson disease diagnostics.

Iveta Zolotová is a professor and a leader of the research group Intelligent Cybernetic Systems at the DCAI FEEI TUKE. Her expertise are: intelligent systems, human-system symbiosis and interaction, digital twin, Industry 4.0/5.0, industrial informatics, applied artificial intelligence, edge AI, industrial IoT, smart health and transportation, computer vision, robotics. She leads many research and innovation projects, and the labs: Think lab: Vison, IoT and Robotics; Intelligent Health Lab, and Smart Industry/Industry 4.0 lab. She is a member of IFIP and a senior member of IEEE.