Data Analytics and Visualisation System for Fall Detection for Elderly and Disabled People*

Parkpoom Wisedsri[1] and Dr. Chutiporn Anutariya[2]

Abstract— **The purpose of this study is to develop a data analytics and visualization system for detecting falls in elderly and disabled individuals. In this study, analytic techniques and intuitive data visualization methods are employed to overcome challenges associated with real-time fall detection. The system is designed to provide immediate alert notifications, enabling prompt assistance. A user-centered approach takes into consideration the unique needs and capabilities of the target audience. Furthermore, the research emphasizes the importance of data visualization in presenting fall-related information in a clear and understandable way, thereby facilitating better decision-making and patient recommendations. By addressing this gap, the study aims to enhance the overall usability of user interface design for fall detection and response systems for elderly and disabled individuals.**

I. INTRODUCTION

According to the United Nations [1], an aging society was characterized by a growing proportion of elderly individuals, specifically those aged 65 and over. These individuals made up 7% or more of the total population. This demographic shift inevitably had an impact on social, economic, and governmental landscapes, leading to challenges in healthcare provision, elder care, and retirement funding.

In response to these complex challenges, the present research aimed to enhance the fall detection platform developed under the TeleHealth Monitoring and Assistive Systems for Elderly and Disabled People project [1], which was sponsored by the Office of the National Broadcasting and Telecommunications Commission. This research effort utilized data from the project to implement a sophisticated data analytics and visualization system, thereby improving monitoring and care for the elderly and disabled population.

The fall detection system, which uses camera-based tracking and deep learning algorithms, was developed to monitor patients and quickly detect fall incidents within vulnerable populations. Despite leveraging cutting-edge machine learning for immediate fall recognition and an emergency API for real-time alert generation, the system lacked an intuitive visualization interface. This shortcoming hindered the effective interpretation of data by healthcare professionals. To address this limitation, the system was redesigned with a focus on usability and alignment with the specific needs of healthcare practitioners. By concentrating on the API and incorporating robust data analytics and visualization techniques, the clarity and utility of the data for healthcare professionals were significantly enhanced.

II. LITERATURE REVIEW

A. Telehealth Technology

This research aims to enhance the quality of care for the elderly and disabled, thereby elevating their overall quality of life. This objective is realized through the utilization of data from Work Package 2.1, which emphasizes the expansion of motion and control analysis. The process includes the installation, integration, and evaluation of hardware packages and analytics. These components constitute an integral part of a comprehensive system. This consists of an edge computing hardware processing system, a RESTful web service and AI Cloud service system, and a web application system for result visualization.

In Work Package 4, a cloud-based, mobile-friendly platform was developed specifically for analyzing, visualizing, and managing data that resulted from Work Package 2.1. This research is focused on data collection from the daily activities of the elderly and disabled.

The system was designed for to three primary groups of intended users. Families and caregivers and healthcare professionals could access fundamental data for care planning. Healthcare professionals could leverage data analysis results for clinical decision-making and sophisticated medical research models. Developed with a user-centered approach, the system affords functionalities such as patient movement filtering by time, emergency notifications, incident analysis based on environmental factors, fall diagnoses, and recommendations. This platform serves to facilitate research models and clinical decisions for medical personnel, simultaneously furnishing caregivers and family members with vital data and analysis results.

B. Data Analytic

In data analytics, raw data is examined for insights. Data analytics in healthcare [2] provides a comprehensive analysis of current and historical data within the healthcare field. Its primary purpose is to forecast trends, improve disease management strategies, and enhance patient outreach initiatives. The analytical approach enhances patient care quality, facilitates accurate clinical data interpretation, optimizes diagnostic processes, and refines overall healthcare business operations. Integration with business intelligence suites and

*This study was supported by a research grant from The National Broadcasting and Telecommunications Commission, served by its operating body the Office of the NBTC, is an independent regulatory agency of Thailand (Contract No. B64-2-(2)-005).

[1]Parkpoom Wisedsri Doctoral student at School of Engineering and Technology Asian Institute of Technology Pathum Thani, Thailand parkpoom.wisedsri@gmail.com

[2]Dr. Chutiporn Anutariya Associate Professor at School of Engineering and Technology Asian Institute of Technology Pathum Thani, Thailand chutiporn@ait.asia

[1]"NBTC-telehealth"

data visualization tools further enables real-time information processing, facilitating informed decision-making and actionable insights. Healthcare analytics involves various domains, such as donation management, physician scheduling optimization, claims and client tracking for insurance companies. Healthcare analytics harnesses data to drive meaningful insights, contributing to improvements in healthcare delivery and operational efficiency.

C. User Centre Design

The user-centred healthcare design (UCHD) aims to improve healthcare services by fostering collaboration among patients, healthcare staff, families, and communities. By integrating co-design methods, UCHD aims to understand the experiences and needs of service users and providers, which enables improvements in existing services, innovative service designs, and strategic changes in healthcare. UCHD is particularly useful in addressing the challenges posed by an evolving healthcare landscape. As a result, there is an aging population with complex health issues, an increase in patient self-care responsibilities, and an increase in the expectation of personalized healthcare. With the aid of professional designers, UCHD empowers patient involvement in shaping their healthcare experiences, resulting in a patient-centered healthcare system.

D. Dashboard

Data-driven healthcare dashboards are designed to simplify cognitive loads and enhance comprehension by visualizing data [3]. The design process involves stakeholder engagement, including participation from staff and users, in order to ensure that data is meaningful and aligns with existing workflows. In addition, data collection and upload processes are monitored to ensure data quality and its potential to drive quality improvement initiatives. An example of a dashboard co-designed for aged care settings would provide a snapshot of clients, identify at-risk individuals, and provide evidence-based decision support, ultimately leading to improved client outcomes [4] through real-time monitoring.

E. Comparative Study of Web Development Methods

Web development covers no-code, low-code, and high-code approaches. This text highlights the distinctions between these methods and provides guidance on their appropriate utilization. High-code is a conventional programming approach where proficient developers write code from scratch. Conversely, low-code is perceived as a beneficial tool for businesses seeking rapid and straightforward application creation, while no-code offers even more streamlined development without traditional coding. Overall, the webpage aims to enhance readers' comprehension of web development methodologies. It will also help them select the most suitable approach based on their specific requirements. In Table I, the comparison factors between no-code, low-code, and high-code development are elaborated, offering a detailed perspective on the unique characteristics and potential applications of each approach.

TABLE I
COMPARATIVE ANALYSIS OF LOW-CODE/NO-CODE AND HIGH-CODE DEVELOPMENT APPROACHES

Factors	Low-Code/No-Code	High-Code (Traditional)
Coding Skills Required	Basic knowledge of workflows, APIs, UI/UX	Extensive knowledge of programming languages
Time to Market	Faster, can be reduced by at least 50%	Slower, can take several months for complex apps
Cost	Less expensive, cost savings of 50-70%	More expensive, due to developer efforts and expertise
Maintenance	Less code maintenance efforts and cost	More code maintenance effort, especially if not using clean code
Stability	More stable due to well-tested algorithms	Could have more bugs due to custom coding
Suitable for Type of App	Ideal for basic and moderate level apps	Ideal for advanced and first-of-its-kind apps
Learning Curve	Easier, drag-and-drop functionality	Steeper, requires in-depth knowledge of coding and tech stack
Flexibility	Less flexible, based on platform	More flexible, any feature can be created
Customizability	Less, depends on platform	High, anything can be custom-built

Based on the comparative review, high-code development is recommended for intricate and scalable applications that demand extensive control and significant customization levels. It is well-suited to large enterprises due to its ability to handle complex business logic and substantial data volumes. In the context of the telehealth dashboard system, if it is complex and necessitates significant customization, high-code development would be the preferred choice. Conversely, if the system is straightforward without intricate functionalities or customization needs, no-code or low-code development would be more suitable.

III. METHODOLOGY

The methodology of this study is underpinned by Agile Software Development, an approach to software development that emphasizes flexibility, collaboration, and efficiency [5]. This approach facilitates quality products delivery through continuous interaction and adaptation to change. Agile Software Development extends beyond traditional practices such as pair programming, test-driven development, stand-ups, planning sessions, and sprints. It is a project management approach that breaks the project into manageable phases and underscores the importance of continuous collaboration and improvement [6]. Agile methodologies adopt an iterative approach to software development. Unlike the linear waterfall model, Agile projects are divided into small, manageable units, allowing for additional flexibility and adaptability [7]. This study adheres to the Agile Software Development approach, which emphasizes collaborative engagement, active software implementation, user-centric involvement, and effective change management.

A. Collaborative Engagement

1) Requirement Analysis: The goal was to develop a user-friendly interface that displays data from a fall detection system, following the guidelines of the REST API manual's JSON structure. This process involved extensive analysis of requirements and collaboration with healthcare professionals to better understand both system and user needs. The expected result was an interface that effectively presents system data for an optimized user experience. Requirements were categorized into Mandatory (M), Desired (D), and Optional (O) based on their importance. M requirements were the most critical, D requirements were less significant but beneficial if time allowed, and O requirements could enhance app performance and might be implemented after the main requirements were met.

TABLE II
REQUIREMENT ANALYSIS

	Goal	Priority	F or N
A	Understand patient activity patterns	M	F
B	Understand and respond to the cause of the fall	M	NF
C	Respond quickly and efficiently	M	F
D	Define, Understand, and Prevent Incidents	D	F
E	Respond appropriately	M	F
F	Efficiently detect and prevent incidents	D	F
G	Enhance understanding of activity patterns	O	F
H	Develop better understanding of patient health	M	F

TABLE III
TASK ANALYSIS

Task	Description
A	Filter patient movement by time series
B	Trace back fall detection incidents
C	Receive notifications in case of an emergency
D	Perform an incident analysis that relates to the environment
E	Diagnose falls and classify the cause of the fall
F	Receive recommendations based on frequently occurring incidents
G	View a graph visualization of activities
H	View data visualization of health information

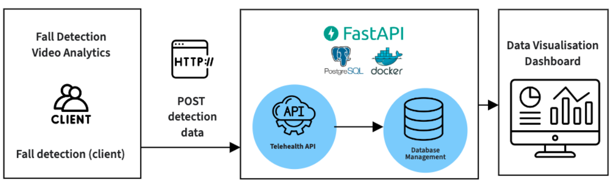

Fig. 1. System Architechture

B. Software Implementation

1) Data Collection: The data used in this study were collected from the TeleHealth project. In order to access the data structure, a JSON API is employed. This API has been derived from a work package dedicated to a machine vision-based fall detection system. There are two types of data generated by the API. In the event that a fall incident occurs involving an elderly patient, the monitoring system immediately transmits fall detection data. The second feature of the system is that it dispatches a dataset every 10 minutes, containing specific properties defined in the following JSON properties.

2) System Architecture: Figure 1 summarizes the framework of backend and front end development. The framework retrieves the data from the API using HTTP request methods. The framework outlined in figure 1 displays an approach designed to the healthcare platform, particularly fall detection.

The data analytical system decodes the JSON data from the API and database, transforming raw data into a user-friendly dashboard format. This UI/UX design emphasizes accessibility, simplicity, and effectiveness, enhancing user-system interaction. Real-time analytics, particularly through the 'Emergency JSON Data Structure,' enable prompt action during critical healthcare events. FastAPI, a high-performance web framework, is utilized for immediate processing and presentation of urgent data, making it suitable for handling real-time emergency JSON data structures.

The second category is the 'Data Summary Structure' that is displayed in [4]. This structure, on the other hand, is responsible for a comprehensive and synthesized view of the collected data. It encompasses an aggregation of data over specific time periods, offering a broader perspective. This allows for insightful trend analysis and informed decision-making based on the accumulated information.

PostgreSQL, an object-relational database management system, is used to handle workloads from small to large applications. It supports creation, retrieval, update, and deletion of database records using Python code, thus managing the Data Summary Structure effectively. By separating real-time data analytic into two structures, the framework addresses both immediate and long-term analytic needs, enhancing its relevance in healthcare. The integration of Fast API and PostgreSQL allows efficient data handling and analysis, promoting immediate response and thorough data scrutiny.

Additionally, the Firebase Real-Time Database can be considered as an alternative option for the project in terms of updating data. Firebase provides a cloud-hosted NoSQL database that allows data to be stored as JSON and synchronized in real-time across connected clients. This real-time synchronization makes it an attractive option for applications that require instantaneous data updates and can be particularly beneficial in healthcare settings where timely access to data is crucial. The use of Firebase Real-Time Database could further enhance the system's responsiveness and adaptability, aligning with the project's goals of immediate response and comprehensive data analysis.

C. User-Centric Involvement

1) Functional and Nonfunctional Requirements: In this study, functional (F) and non-functional (NF) requirements as shown in a table, goal and priority. Table II reveals the overall features of the fall detection system. Moreover, the needs of the healthcare professional, caregiver, and patient are gathered and analyzed. In the table, the terms P and F/N refer to priority and functional or non-functional respectively [8]. As part of the system, a variety of functionality is offered to enhance the quality of health care. Filtering patient movement by time series (A) allows for detailed analysis of patient activity. The ability to trace back and receive immediate notifications in an emergency (B) ensures that swift action is taken in the event of a critical incident. In addition, the system analyzes incidents based on their environment (D), diagnoses falls (E), and provides recommendations (F). Health professionals can make informed decisions with the help of graphic visualizations of patient activities (G) and comprehensive health information (H).

D. Adaptive Planning

The development of healthcare dashboards to support data-driven quality improvements is a form of adaptive planning, as it involves tailoring the display of information to the needs of the users.

1) Visualisation Dashboard: Healthcare dashboards focus on intuitive, user-centered displays, encouraging user-friendly exploration and trend analysis to enhance both patient safety and care quality [9]. These iterative dashboards feature coordinated displays and allow for hands-on data manipulation, spotlighting specific variables within large data sets. In the field of telehealth, these dashboards, supported by a flexible remote monitoring system, work to improve the quality of life for disabled or chronically ill patients. They also act as an innovative training platform for healthcare professionals [10]. It's important to note that this kind of dashboard not only makes data analysis more efficient but also boosts the quality of care provided to patients, whether in traditional healthcare settings or through telehealth.

IV. Discussion

A. Data Structure

1) JSON Data Structure: In the following section, the specifics of data structures generated by an API are discussed. The API is designed to provide data essential for monitoring patient health. The first data structure as shown in table IV is initiated when the patient experiences an urgent fall. This crucial system is designed to immediately convey the state of the patient at the moment of a detected fall. This facilitates rapid emergency services response. The data is displayed in the figure 2.

The second data structure in table V provides a summary of the patient's activities. This data is accumulated and updated at ten-minute intervals, providing a continuous flow of information regarding the patient's actions and overall mobility throughout the day. This wealth of data tracks patient health status and daily routines. The data is display in the 3.

The following figures provide a detailed visual representation of the two data structures:


```
{
  "pateint_name": "Anong Sasithorn",
  "activity_type": "Stand",
  "camera_id": "Hallway",
  "house_id": "1",
  "datetime": "2023-02-22 12:53 PM",
  "Image": "encoded-base64-string-converted-from-
    image-file"
}
```


Fig. 2. JSON Activities list Data Structure


```
{
  "activity_stat": {
    "total_house": "3",
    "house_0001": {
      "house_id": "1",
      "total_detection": "3",
      "detection_list": [
        {
          "activity_type": "walk",
          "timestamp": "2023-02-22 12:53 PM",
          "family_member": "Anong Sasithorn"
        }
      ]
    }
  }
}
```


Fig. 3. JSON Activities Summary Data Structure

These diagrams will offer a clearer understanding of how these data structures function within the context of patient monitoring and health management. Further examination of this data and its potential applications will be discussed in subsequent sections.

TABLE IV
Patient Activity Data Structure

Field	Description
Patient Name	The name of the patient (e.g., Anong Sasithorn)
Activity Type	The type of activity being performed by the patient (e.g., Stand)
Camera ID	The identifier of the camera capturing the activity (e.g., Hallway)
House ID	The identifier of the house where the activity was detected (e.g., 1)
Datetime	The timestamp when the activity was detected (e.g., 2023-02-22 12:53 PM)
Image	An encoded string representing an image file

B. Prototype Design

1) Introduction to Prototype Design: The result of the prototype, specifically in terms of the user interface, is depicted in Figure 4. This figure elucidates the comprehensive user interface design. Additionally, the following nine main pages, identified as key findings, are illustrated to demonstrate the functionality of each individual page.

TABLE V
HOUSE ACTIVITY STATISTICS DATA STRUCTURE

Field	Description
Total House	Total number of houses (e.g., 3)
House ID	Unique identifier for each house (e.g., house_0001)
Total Detection	Total number of detected activities in the house
Detection List	List of activity detections

TABLE VI
DETECTION LIST DATA STRUCTURE

Field	Description
Activity Type	The type of activity detected (e.g., walk)
Timestamp	The time when the activity was detected
Family Member	The name of the family member who performed the activity

Page 1: Data in structured graph format. It presents key patient data such as `patient_name`, `activity_type`, `camera_id`, `house_id`, `datetime`, and `image`. The image data is an encoded base64 string.

Page 2: Dedicated to activity monitoring, this page features a scatter plot graph that represents activities over time, marked by a red line to indicate the timeline. It also includes a filter module for camera selection, house ID, and date range selection.

Pages 3-4: These pages exhibit a timeline graph identical to Pages 1 and 2 but incorporate a pop-up notification system for fall detection alerts. They also provide a 'track back' analysis pattern within the system to investigate historical activity data.

Pages 5-6: Time series data is displayed along with a table of the patient's historical activity, providing a more comprehensive perspective on the patient's previous behavior.

Pages 7-9: These pages probe into the patient's historical data, in conjunction with personal health metrics that the camera might not capture, such as falls in places where cameras are not installed. A potential future feature on these pages could be a data recommendation module curated by medical experts.

Based on the iterative design process and feedback from users and collaborators, certain pages and features within the prototype have been identified as more critical and advantageous than others. Figure 4 illustrates the overall prototype design of the entire system.

C. Front-end Prototype Design and Finalization

The Software Usability Measurement Inventory (SUMI) is a standardized method that evaluates software from a user's perspective, focusing on five key aspects: efficiency, control, affect, helpfulness, and learnability [11]. These principles guide the development of systems that not only meet technical requirements but also align with the practical needs and preferences of end-users. The evaluation of the prototype, informed by SUMI principles, provides a comprehensive understanding of the system's capabilities and potential improvements.

1) Efficiency and Control: By using scatter plot graphs, the system should be able to display information from multiple cameras on a single page. Users have control over the information they view due to this simultaneous representation of various activities.

2) Affect and Usefulness: In addition to information captured by the camera, the system should incorporate data related to the specific locations of recorded activities. This location-specific data can enhance the user's psychological response to the system and increase its effectiveness.

3) Efficiency and Learnability: The system should transparently display information about those recognizing fall emergency notifications, including the number of people involved in the alerting process. This feature can improve emergency response efficiency and make the system easier to learn and use.

4) Control and Affect: The system should issue notifications when users are detected in areas with potential fall risks due to obstacles or unexpected objects. This increases user control and positively affects their interaction with the system.

5) Helpfulness and Efficiency: The user profile should incorporate a module to record data about incidents outside the camera's detection zone for privacy reasons. This data can provide valuable insights to doctors for predicting future fall risks, making the system more helpful and efficient.

6) Control and Learnability: While some experts suggest a data recommendation feature, it must be implemented carefully considering privacy and individual differences. This feature, if designed thoughtfully, could serve as a valuable tool for doctors in customizing patient data. This would enhance user control and system learnability.

D. Usability Principles and Evaluation

The prototype is intended for use by professional healthcare teams. Evaluation surveys have been provided by experts in medical health with extensive experience. These evaluations elucidate the extent to which the prototype fulfills mandatory requirements in terms of efficiency, control, affect, helpfulness, and learnability. In addition to these evaluations, open-ended questions further enrich the survey by providing qualitative feedback. This comprehensive approach ensures a thorough and multi-faceted assessment of the system's capabilities and potential improvements, in line with the principles of usability testing using questionnaires.

E. Contribution

In summary, this research focuses on developing a prototype for a fall detection system tailored to the needs of healthcare professionals and caregivers of the elderly and disabled. The design process follows a user-centered approach, starting with a comprehensive understanding of user needs through techniques like interviews and observations. Findings from this initial research inform the system's design, highlighting crucial functionalities such

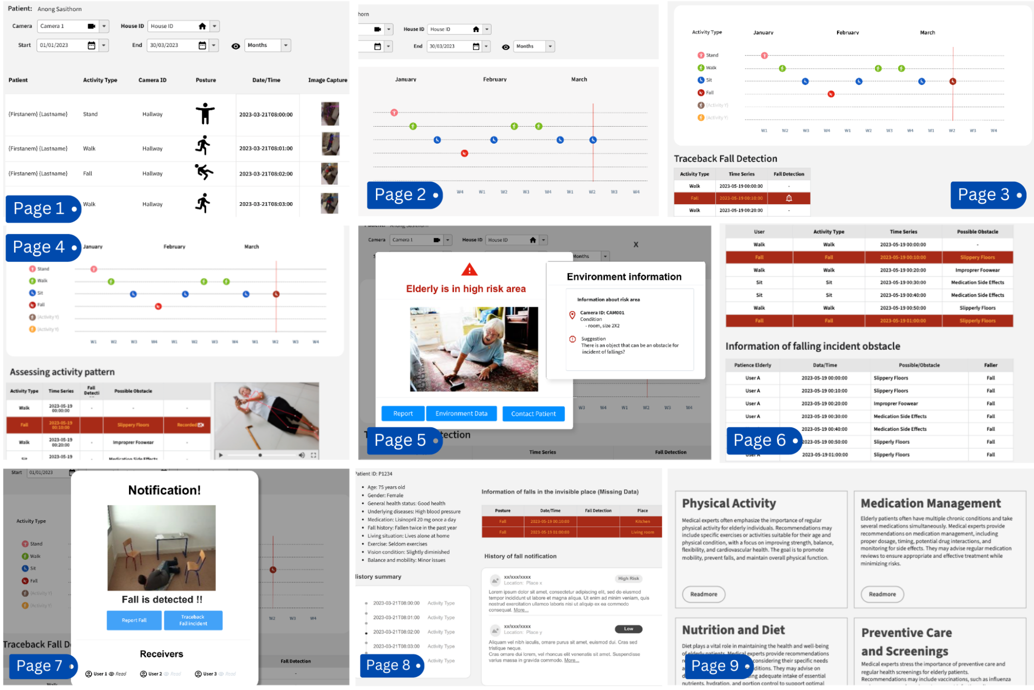

Fig. 4. The Overall of the Prototype Design

as real-time alerts, fall incident analysis, and pattern-based recommendations.

The prototype's development is accompanied by user testing to assess its effectiveness, efficiency, and user satisfaction. Feedback from these tests drives iterations and improvements, ensuring the system provides an intuitive and user-friendly experience. This iterative design, testing, and refinement process continues until the system adequately addresses user needs and expectations. The research culminates in a fall detection system that combines advanced data analytics, visualization techniques, and user-centered design, enhancing the user experience. This research's scope in user-centered design can serve as a model for future studies, emphasizing the importance of prioritizing user needs in creating impactful and effective systems.

V. Acknowledgment

This study was supported by a research grant from The National Broadcasting and Telecommunications Commission, served by its operating body the Office of the NBTC, is an independent regulatory agency of Thailand (Contract No. B64-2-(2)-005).

References

[1] United Nations, Department of Economic and Social Affairs, Population Division, *World Population Ageing 2019*, United Nations, 2019,

[2] Sisense, "Healthcare Analytics Basics," Available at: `https://www.sisense.com/glossary/healthcare-analytics-basics/`, Accessed: 29 May 2023.

[3] K. Ludlow et al., *Co-designing a dashboard of predictive analytics and decision support to drive care quality and client outcomes in aged care: a mixed-method study protocol*, *BMJ Open*, vol. 11, no. 8, 2021, doi: 10.1136/bmjopen-2021-048657.

[4] N. Alvarado et al., *Analysis of a Web-Based Dashboard to Support the Use of National Audit Data in Quality Improvement: Realist Evaluation*, *J. Med. Internet Res.*, vol. 23, no. 11, 2021, doi: 10.2196/28854.

[5] Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Kern, J. (2001). Manifesto for agile software development.

[6] Schwaber, K., & Beedle, M. (2002). Agile software development with Scrum. Prentice Hall.

[7] Larman, C., & Basili, V. R. (2003). Iterative and incremental developments. a brief history. Computer, 36(6), 47-56.

[8] P. Wisedsri, *Mobile Application For Workout Planning Based on Weather and Air Quality*, MSc Dissertation Project (COM6906), University of Sheffield, September 2020. A report submitted in partial fulfilment of the requirements for the degree of MSc in Computer Science in the Department of Computer Science. Supervisor: R. Dimitrova.

[9] Ratwani et al., *'Connecting the dots': leveraging visual analytics to make sense of patient safety event reports*, *Journal of the American Medical Informatics Association*, vol. 22, no. 2, pp. 312–317, 2015, doi: 10.1136/amiajnl-2014-002963.

[10] Albert, Arteta and de Mingo López, Luis Fernando and Gómez Blas, Nuria, "Tele-Treatment Application Design for Disable Patients with Wireless Sensors," *Applied Sciences*, vol. 10, no. 3, pp. 1142, 2020, doi: 10.3390/app10031142.

[11] J. Kirakowski, "The Software Usability Measurement Inventory: Background and Usage," in *Proceedings of the European Software Quality Week*, Brussels, Belgium, 1998.