

PAPER

Assistive Technology for Inclusive Education: A Single-Case Study of a Child with Spastic Tetraplegia, Visual and Motor Disabilities in Ecuador

Milton Campoverde-Molina¹ ,
 Juan-Carlos Cobos-Torres² ,
 Sergio Luján-Mora³ , Carlos
 Paúl Ortiz-Alvarez⁴ , Eugenio
 Esteban Palomeque-Zambrano⁴

¹Unidad Académica de Informática, Ciencias de la Computación, e Innovación Tecnológica, Grupo de Investigación Simulación, Modelado, Análisis y Accesibilidad (SMA²), Universidad Católica de Cuenca, Cuenca, Ecuador

²Unidad Académica de Posgrado, Carrera de Ingeniería Eléctrica, Grupo de Investigación en Radiación Visible y Prototipado GIRVyP y Grupo de Investigación en Sistemas Embebidos y Visión Artificial en Ciencias Arquitectónicas, Agrícolas, Ambientales y Automáticas (SEVA4CA), Universidad Católica de Cuenca, Cuenca, Ecuador

³Departamento de Lenguajes y Sistemas Informáticos, Universidad de Alicante, Alicante, España

⁴Universidad Católica de Cuenca, Cuenca, Ecuador

mcampoverde@ucacue.edu.ec

ABSTRACT

In this study, we designed and developed an adapted keyboard and software for learning to write for an Ecuadorian child with spastic tetraplegia, visual and motor impairment using a computer. The method used in this study was the adaptation of the Human Activity Assistive Technology (HAAT) model. First, functional and non-functional hardware and software requirements were analyzed. Then, the adapted keyboard and software were designed based on the analysis of the requirements. The adapted keyboard consists of 10 keys: three circular keys (Esc, Character, and Select Character), two rectangular keys (Space and Enter), and five triangular keys (four scroll arrows and Erase). The software is compatible with any screen reader, such as JAWS, due to the child's visual impairment and allows writing on a computer through an auto-complete of words previously stored in a database. Finally, the keyboard and software were developed based on the previously elaborated designs and considerations of the child with disabilities. Several tests were developed with the JAWS screen reader to corroborate the accessibility and functionality of the adapted keyboard and software. In conclusion, the keyboard and software helped to improve the written learning. The target audience for this paper is researchers, families, educators, and students with disabilities.

KEYWORDS

assistive technology, disabilities, inclusive education, keyboard, software, writing

1 INTRODUCTION

Assistive technologies can help people with disabilities use computers to perform daily activities independently and communicate with others [1]. Assistive technologies [2] include appliances, devices, equipment, processes, services, software, and ergonomic adaptation. These technologies enable people with disabilities

Campoverde-Molina, M., Cobos-Torres, J.-C., Luján-Mora, S., Ortiz-Alvarez, C. P., Palomeque-Zambrano, E. E. (2025). Assistive Technology for Inclusive Education: A Single-Case Study of a Child with Spastic Tetraplegia, Visual and Motor Disabilities in Ecuador. *International Journal of Online and Biomedical Engineering (iJOE)*, 21(11), pp. 45–65. <https://doi.org/10.3991/ijoe.v21i11.56109>

Article submitted 2025-04-17. Revision uploaded 2025-07-11. Final acceptance 2025-07-11.

© 2025 by the authors of this article. Published under CC-BY.

to overcome social barriers. In addition, accessible and safe technology achieves autonomy and independence to participate fully in society without affecting health.

Sensory, physical, or mental disabilities caused by genetic or traumatic lesions prevent people from interacting with their environment [3]. However, assistive technologies [4] have transformed the lives of many people with disabilities through hardware and software mechanisms that allow them to interact with computers. Hardware and software mechanisms can be used through the senses, such as sight, hearing, smell, taste, or touch, and any fine or gross motor skills, such as the jaw or the wrist, among others. In addition, various types of adapted keyboards have been developed around the world for people with multiple disabilities [5].

According to [6, p. 51], “students with disabilities are often challenged when using a conventional keyboard with visual, cognitive, or motor disabilities.” The most common problems for students with severe disabilities are: finding the keys to be pressed and remembering the sequence of keys, their location, and their size [6]. Therefore, the development of a specific keyboard for people with severe disabilities places users in a more active position in the educational and social sphere through the use of a computer. These adapted keyboards allow students with disabilities to enter words, phrases, and numbers at the touch of a key. In addition, they can be adapted and configured according to the movement limitations of students with disabilities. The adaptation of keyboards can take many forms, such as ergonomic, presentation, number of keystrokes per character, location of keys and sounds, among others [7].

Various types of adapted or virtual keyboards [8], [9], [10], [11], [12] have been developed that have contributed to the interaction of people with disabilities with a computer. In addition, they help in the teaching and learning process for students with disabilities [13], [14]. In light of the fact that education is located within a digital age, in which technologies offer individuals with disabilities novel avenues of inclusion in contemporary society. Therefore, it is imperative to consider the potential of these technologies to facilitate equitable access to education.

According to [15, p. 42], “inclusive education aims to provide equitable learning opportunities for all students, regardless of their diverse backgrounds and abilities.” Therefore, the main challenge to achieving inclusive education in the classroom is for teachers to adopt inclusive and pedagogical values [16]. Inclusive education is not limited to replicating special education strategies but to innovating and adapting teaching methods to meet the diverse needs of all students [17]. However, there are still many barriers to achieving inclusive education [18]. One barrier is the absence of assistive technology that meets the specific needs of people with multiple disabilities. In addition, assistive technology has a significant problem: it is expensive and lacks personalization. Furthermore, the majority of studies on child development are conducted in high-income countries, and as a result, the findings from low- and middle-income countries (LMICs) remain limited [19].

If designed and implemented correctly, assistive technologies can become a powerful ally in the home and in education for many families with children with disabilities. Therefore, this study aims to develop two assistive technologies: an adapted keyboard and software, making use of the Human Activity Assistive Technology (HAAT) model [20] so that a child (student) with spastic tetraplegia, visual and motor impairment can type using the computer, through the range of motion of the upper limbs and posture.

The adapted keyboard was designed using the Rapid Upper Limb Assessment (RULA) [21], [22] method to assess the different postures of the human body in a seated position and the reach angles of the upper limbs. In addition, it determined whether ergonomically the keyboard is not detrimental to the health of the student. The layout of the keyboard should follow the range of motion of their limbs, reach, and motor skills. The keyboard operates with commands, since the student cannot type with his fingers. The software allows the child to learn to write using the Job Access with Speech (JAWS) screen reader and the adapted keyboard on a computer. To do so, the child must press a key on the adapted keyboard that auto-completes existing words in a database. The teacher must have previously recorded the set of words that the child is to learn. In addition, the tasks to be performed by the child, using the software, are previously planned by the teachers and executed with the teacher. The adapted keyboard and software are developed using the requirement engineering [23]. These were defined according to the specifications of the motor and communication teachers and family members.

2 RELATED WORK

The integration of assistive technologies, such as adapted keyboards, software, etc., for people with multiple disabilities is a research topic that addresses the diverse needs of users with various motor, visual, and cognitive disabilities. Adaptive keyboards and alternative input methods have been developed to improve accessibility [24] and usability for these individuals, significantly improving their interaction with digital devices. However, research concerning the use and development of assistive technologies in LMICs is lacking.

In 2013 [25], a program promoting the independent choice behavior of three children with cerebral palsy and multiple disabilities was evaluated. The program was based on assistive technology (input devices/sensors, computers, and stimulus selection according to a binomial criterion) and learning principles. The first objective was to provide participants with choices among three categories (food, drink, and leisure) and to request a specific item from four in each category, adopting a procedure that minimized involuntary choices. The second objective was to follow up on the program's effects on mood (happiness index) through detectable signs. The results showed increased engagement and happiness during the intervention phases. In addition, psychological and educational implications were analyzed.

In 2014 [26], a study stated that dynamic keyboards represent an innovative solution. This study found that dynamic keyboards can significantly increase text entry speed for people with functional tetraplegia compared to traditional QWERTY keyboards. In addition, the study highlighted that such systems reduce the number of selections required and the latency between selections, thus improving the overall user experience. The potential of adaptive technologies [7] to meet the needs of users with motor impairments could reduce interaction costs and improve text input speed.

In 2020 [1], a word prediction method based on a hidden Markov model was produced that allows keyboard emulation software to predict words so that children with disabilities can type texts faster. The keyboard emulator, the construction and processing of a corpus, and a word prediction algorithm were developed. Children with different cognitive profiles had to produce a text and write it twice: the first

time with free writing and the second time using the word prediction of the virtual keyboard. The results indicated that the word prediction of the keyboard emulator software reduced the typing effort. In addition, all volunteers required fewer clicks to perform the task. People with disabilities (especially children) could use the system and demonstrate their knowledge and skills.

In 2022 [27], one of the most prominent approaches is using ability-based keyboard generation methods, which adapt keyboard layouts to the specific abilities and movement patterns of users with motor disabilities. For example, the authors demonstrated that customized keyboard configurations could improve the communicative efficiency of people with dexterity impairments by optimizing key layouts based on the user's specific cursor control abilities. This method adapts to users' unique movement patterns and reduces the time and effort required for text input, which is especially beneficial for people with limited motor control.

3 METHODOLOGY

Cook and Polgar's HAAT model [20] is a theoretical framework for implementing assistive technology. The HAAT model [28] describes the person performing an activity in a context using assistive technology. The model's components are the activity, the human being, the assistive technology, and the context [29]. Assistive technology is at the center of this model, but it cannot be considered independently of the other components.

Considering that our study focuses on creating assistive technology, the HAAT model is used to develop the adapted keyboard and software for a child with multiple disabilities to learn to write on a computer. Figure 1 shows the HAAT model and briefly describes each component (human, assistive technology, and activity), their interrelationship, and the steps. These components interact with the context [30], which is the all-encompassing component of the model. The HAAT model consists of four components:

- **Context.** This component [29] includes the physical, social, cultural, and institutional environment that determines the use of assistive technology. In our study, the context represents the inclusive education of a child with multiple disabilities.
- **Human.** This component [29] represents the individual's physical, emotional, cognitive, and sensory abilities. In our study, the human is a child with spastic tetraplegia and visual and motor impairment. In addition, this component presents the deepening of the problem.
- **Assistive technology.** This component [29] is described as a device that enables an activity to be performed, considering the person/technology interface that influences the selection and use of assistive technology. In our research, assistive technology is the creation of an adapted keyboard and software for learning to write using a computer.
- **Activity.** This [29], in the HAAT model, considers the task(s), task demands, and contextual factors. In our study, the activity consists of testing the adapted keyboard and software with a child with spastic tetraplegia, visual, and motor impairment using a computer to learn to write.

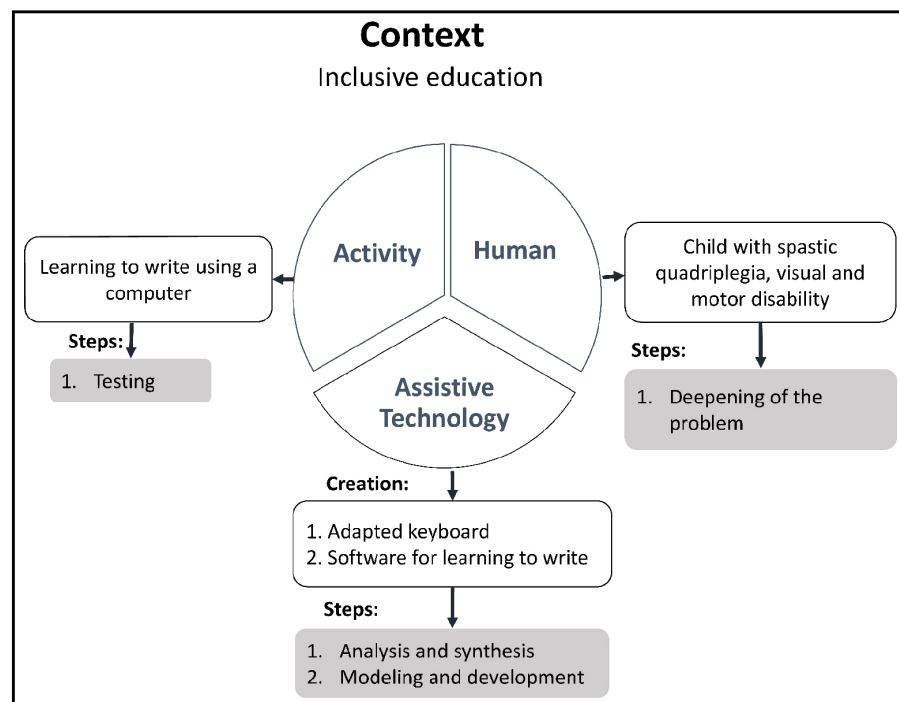


Fig. 1. HAAT model for the development of the adapted keyboard and a software: Adapted from [20]

4 RESULTS

This section describes the results obtained from the application of the HAAT model. The subsections detail context, humans, assistive technology, and activity.

4.1 Context

In 2015 [31], the member states of the United Nations adopted 17 Sustainable Development Goals (SDGs) as part of the 2030 Agenda for Sustainable Development, which sets a plan to achieve the goals in 15 years. The 4th SDG highlights “Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all,” and its “Target 4.a Build and upgrade education facilities that are child, disability, and gender sensitive and provide safe, non-violent, inclusive, and effective learning environments for all.” According to the United Nations Educational, Scientific and Cultural Organization (UNESCO) [32], the changes brought about by the rapid development of information and communication technologies (ICT) not only create diverse opportunities for humanity but also pose new challenges.

Through technology [33], people with disabilities can participate more actively in learning and improve communication and interaction with peers and teachers. Technology can provide them with better and more efficient access to the curriculum while developing their independence and social inclusion. In addition, technology can facilitate personalized learning and improve skills through reinforcement exercises. Therefore, inclusive education [34] for students with disabilities is often not possible without their access to appropriate assistive technology,

as the barriers to their education are often from their environment. For the development or adaptation of assistive technologies, each child's specific needs must be considered, and a forward-looking view of the use of assistive technology must be taken to ensure that school-age children begin to benefit from it as early as possible.

4.2 Human

The problem was defined based on several interviews with motor therapy and communication teachers at a Special Education Unit in Cuenca, Ecuador. This institution serves students with hearing and visual impairments, ranging from early stimulation programs to tenth grade, as well as children with various physical or intellectual disabilities.

The case study focuses on a 10-year-old male child with spastic tetraplegia, along with visual and motor impairments, who is currently unable to use a computer for writing tasks. Therefore, it is necessary to develop an adapted keyboard tailored to the movement range of his upper limbs and his seated posture in a wheelchair. Spastic tetraplegia has significantly limited the mobility of his upper limbs, impairing his motor skills. The combination of blindness and a lack of fine motor control prevents him from using conventional computer input devices or existing assistive technologies.

To design an effective adapted keyboard, several physical factors must be considered, including the reach of each arm, wrist rotation, wrist tension, and both fine and gross motor accuracy. The keyboard should also include a protective polymer cover to prevent involuntary key presses caused by muscle contractions. Additionally, it should have a physical binding or recessed area to prevent the device from being displaced or falling due to unintentional movements. The key size should be larger than that of standard keyboards to accommodate limited precision.

Complementary software is also required to allow the child to interact with both the adapted keyboard and the computer. This software should support word auto-completion based on the initial letter selected in order to enhance typing speed. Furthermore, it must be compatible with the JAWS screen reader and function properly on both Windows and macOS operating systems. The software associated with the keyboard should allow the person with a disability to use the computer [35]. Furthermore, the software should possess the capacity to facilitate communication with a person with a disability, allowing the provision of guidance and assistance in the completion of computer-based tasks.

4.3 Assistive technology

The development of assistive technologies arose from the need for a low-cost solution, a homegrown solution as opposed to more expensive commercial solutions. In this subsection, the evaluation, synthesis, modeling, and development of the adapted keyboard and software are performed, starting from the user's needs. This step took place over a period of three months, during which numerous meetings and discussions were held with motor therapy and communication instructors, as well as with the user's family members. Figure 2 shows the timeline for the analysis and synthesis, modeling and development, and testing of the adapted keyboard and software.

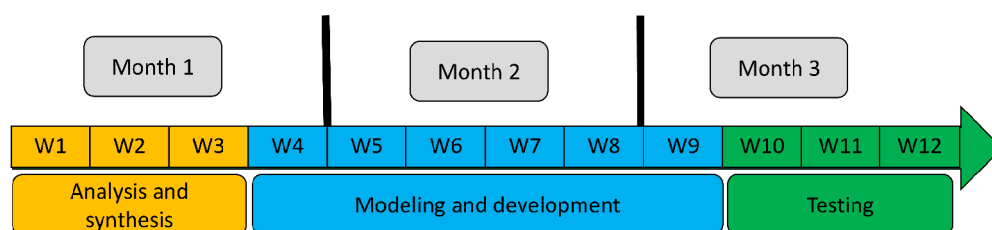


Fig. 2. Timeline of the analysis and synthesis, modeling and development, and testing of the adapted keyboard and software (*W = Week)

Analysis and synthesis. In this step, we determine the factors that should be considered for modeling and developing the adapted keyboard and software. The keyboard and software must be designed according to the limitations of the child with a disability. Therefore, brainstorming established a hierarchy of constraints based on observations and interviews with users, caregivers, and the user's family members. Figure 3 presents the factors (ergonomic, reduced number of keys, word auto-complete, keys with increased dimensions, key protection, and keyboard fastening) to be considered in the requirements, modeling, and development of the adapted keyboard and software.

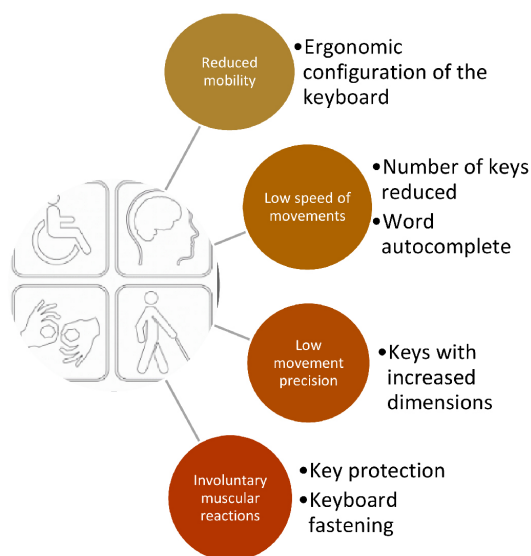


Fig. 3. Factors to consider in the modeling and developing of the adapted keyboard and software

For requirements acquisition, requirements engineering [36], [23] was used, which involves the representation and analysis of requirements and processes. In addition, requirement engineering [37] proposes specific steps and phases to identify user requirements, and software and hardware functionalities and their interaction with the end user are analyzed. In our study, the requirements were acquired through interviews directed to the motor therapy and communication therapy teachers and the user's relatives to develop the adapted keyboard and software. The requirements obtained are presented below.

Software. Software requirements can be classified [38] as functional, i.e., explicit product features or functions, or non-functional [39], i.e., implicit product quality criteria. Functional requirements (login, register user, register word, character input, word predictor, and user reports) and non-functional requirements (ease of use, accessibility, and performance) are presented below.

Functional requirements:

- **Login.** Teachers can log in to the software using their credentials.
- **Register user.** The software administrator can register teachers and students in the system.
- **Register word.** The teacher can record new words as a dictionary, and the word predictor can use these words for auto-completion.
- **Character input.** Using the adapted keyboard, the student can practice writing. The characters are entered in a component of the text area.
- **Word predictor.** As the student types the characters of a word in the text area of the module to learn to write, the software should display the words beginning with those characters for auto-completion.
- **User reports.** Teachers can obtain reports of the activities carried out by the students that each teacher has recorded.

Non-functional requirements:

- **Ease of use.** Due to motor impairment and lack of hand movement, the software must have user-friendly interfaces [40]. In addition, navigation is done with the keyboard, so the emphasis should be on the order and a low number of components and interfaces.
- **Accessibility.** User interface components must be accessible for use with screen readers.
- **Performance.** The software must be cross-platform and installable on operating systems such as Windows, Linux, and Mac. It must also require minimal hardware resources for installation and have a fast word predictor display speed.

Hardware. Similarly, the functional requirements have been considered when constructing the keyboard hardware. The functional requirements (ergonomics, key protection, larger keys, and keyboard fastening) of the keyboard hardware are presented below:

- **Ergonomics.** The keyboard must meet ergonomic standards due to the limited range of mobility of a person with spastic tetraplegia. Consideration is given to the range of each arm, wrist rotation, wrist tension, and fine and gross accuracy.
- **Key protection.** The keys should have a protective polymer sleeve to prevent muscle retractions due to accidental key presses.
- **Larger keys.** Due to the lack of precision, the keys must have dimensions larger than standard to facilitate keystrokes.
- **Keyboard fastening.** The keyboard's design must consider muscle retractions to prevent it from moving or falling.

Modeling and development. In this step, the body characteristics relevant to the ergonomic design of the adapted keyboard and software were compiled. These characteristics include position, upper limb size, motricity, and the ability to perform a job. The adapted keyboard and software are modeled at the process level in this step. These models comprise the functional, operational, and content processes that guide the development of software and hardware.

Software modeling. This subsection uses case diagrams to determine the actors' relationship with the system processes. For this purpose, a use case diagram was created with the system's functional requirements. Using case diagrams [41]

allows developers to analyze and understand the functional requirements of the system. It also enables communication between developers and users to match the system to their needs. Use case models include actors (external entities interacting with the system) and use cases (sequences of actions initiated by actors representing complete functionalities).

Also, a class diagram was created to determine the system's structure and to model its classes with attributes, methods, and relationships between objects [41]. With the class diagram, we created a relational database containing tables called entities and their relationships.

Layered architecture. The software for learning to write on a computer was developed using four layers: the presentation layer, the business layer, the persistence layer and the data layer. The layered architecture presented in Figure 4 is used to develop the software for learning to write.

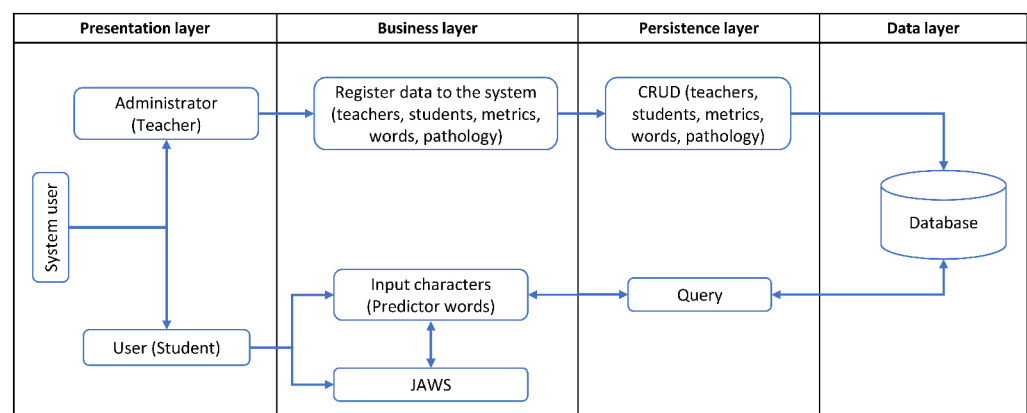


Fig. 4. Layered architecture of the software for learning to write

The layered architecture separates the software functionalities into layers, where each performs specific tasks. The description of each of the layers is presented below:

- **Presentation layer.** This layer manages the user interface and the logic of user communication with the software (teachers, students, metrics, words, and pathology), considering the accessibility of its components for interaction with JAWS.
- **Business layer.** This layer executes the software rules according to the user's needs. Therefore, metrics based on the students' pathologies are applied here.
- **Persistence layer.** This layer contains the code to access the data layer, such as the database connection and SQL statements for insert, update, delete, and select (teachers, students, metrics, words, and pathology).
- **Data layer.** This layer allows the data entered into the software to be stored in a database.

Software development. The software for learning to write used a layered architecture. This architecture improved the user experience by facilitating the distribution of resources and seamless communication between the software, JAWS, and the user. This software was developed using the Java programming language and a MySQL database.

To ensure software accessibility, one of the responsibilities of programmers is to provide accessible names and descriptions [42] for the elements that make up the interface. For example, JButtons and JTextFields have been used because they allow

tabbed navigation and have the properties “Accessible Name” (is the text of the object name) and “Accessible Description” (is a text description of the purpose of an object) [43]. Screen readers, in our case JAWS, read the description of buttons (JButtons) and text fields (JTextFields) using these properties.

The software for learning to write on a computer was developed using the waterfall methodology, which includes requirements analysis, development, implementation, and testing [44]. The software consists of the following modules:

- **Login.** This allows users to register with their credentials and enter the system using their username and password.
- **Home.** This interface of software for learning to write with an adapted keyboard for people with motor and visual disabilities welcomes and provides information about the software.
- **Administration.** In administering the software for learning to write, the user can enter, update, or delete teachers, users (students), pathologies, and words or generate the user’s learning reports.
- **Writing.** This uses an algorithm [45] to auto-complete words using the adapted keyboard and JAWS in a text field. This algorithm performs a binary search of the characters entered in the text field against words previously stored in the database Words table. The words were then loaded from the database into an array list for use with the AutoComplete class.

User interface tests were performed, evaluating the layout of the elements in the graphical interface of the forms and the use of the JAWS screen reader with the software. In addition, the software was tested with the end user to determine its functionality and effectiveness in writing. Figure 5 presents screenshots of the software login interface, the home interface, the administration interface (teachers, user registration, pathologies, and words), and finally the writing learning interface.

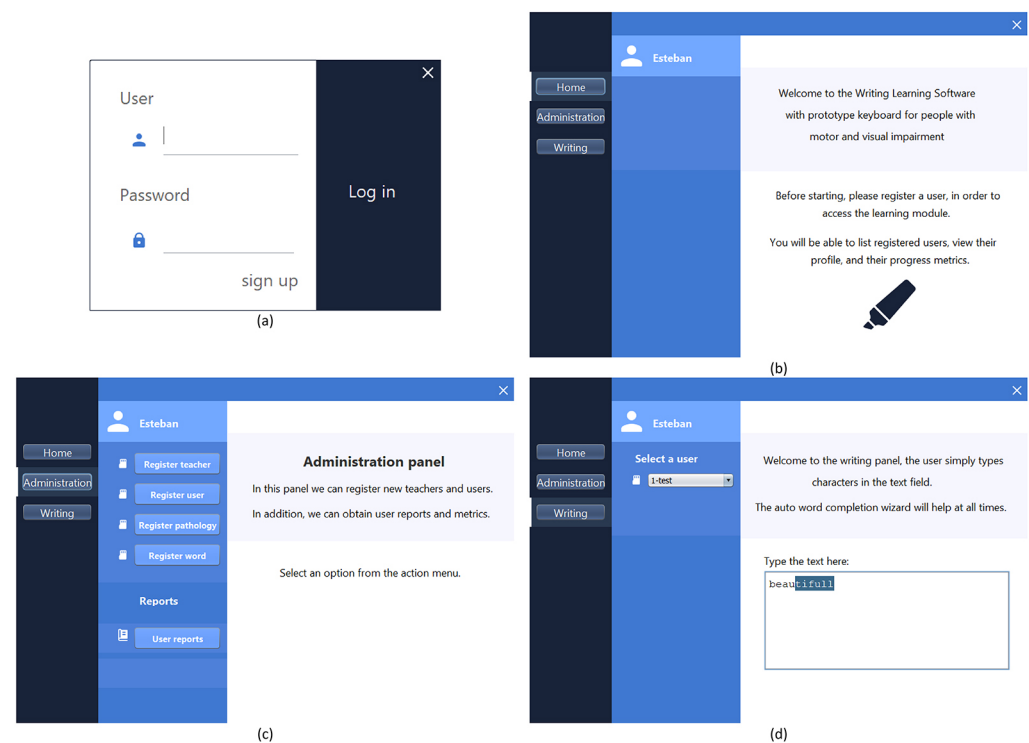


Fig. 5. Software for learning to write: (a) Login; (b) Home; (c) Administration; (d) Writing

Hardware modeling. The adapted keyboard was designed using the RULA method [22]. This method allowed for analyzing the child's position from different angles without harming his health. Therefore, the seated position was considered the most appropriate for the child's motor disability. In addition, measurements of the child's limbs in length and angular data were obtained to establish the range of action on the keyboard. The key arrangements were made according to the range of action of the limbs, the reach, and the motor abilities. The adapted keyboard operation was in command mode, as the child could not type with his fingers. In addition, the child was observed to have a limitation in his left shoulder and wrist.

Considering the sitting position and angular data, the adapted keyboard layout was designed, and it was possible to establish the maximum reach of the upper limbs over the keyboard. The data were geometrically calculated with the length of the left arm at a flexion of 20° , and the law of cosines obtained the opposite cathetus. The result was 7.64 cm plus 14 cm of the forearm length. Furthermore, applying the law of cosines, the opposite cathetus was obtained with the wrist flexed 80° ; the result was 1 cm. Therefore, the reach of his left limb was 22.64 cm over the keyboard. In addition, the same technique was used for his right upper limb, which reached 27.64 cm. Considering that the human shoulder has an internal rotation of 90° , and with the previously established ranges, an arc is obtained, which would mean the range of action of his limbs on the keyboard. Figure 6 presents the design and dimensions of the adapted keyboard and its keys.

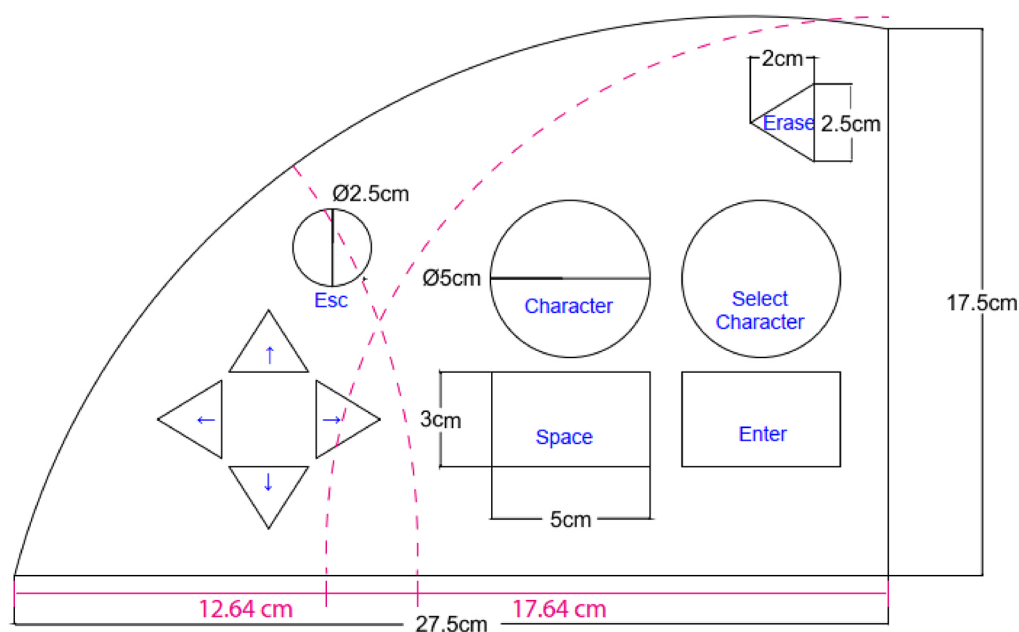


Fig. 6. Design and dimensions of the adapted keyboard and its keys

In addition, the layout and size of the keys were made, considering the stress on the tendons of the child's left wrist, which remained flexed 80° , reducing its radius of action. The operation of the adapted keyboard is as follows: When the character key is pressed, the character "a" is written on the screen; if the character key is pressed again, "a" is replaced by the following letter corresponding to alphabetical order, in this case "b," etc.; this is done by the central circular keys. The tab key accepts the word entered; the space key and the enter key correspond to the rectangular keys.

The delete key is located on the top right. In the left sector are the arrow and escape (Esc) keys.

3D design of the keyboard case. The case and keys of the matrix keyboard were designed using Shapr3D¹ software, a fully 3D design application that allows users to generate 3D models quickly and accurately. 3D printing technology was used to print the design, allowing the size and shape of the models to be manipulated during the prototyping phase and lowering the cost of new parts [46]. There are several suppliers of plastic materials in the 3D printer market. Objects can be produced with minimal waste, reducing the required raw materials. Therefore, the parts were printed with an Anycubic 4Max Pro 2.0 3D printer using ABS filament. As shown in Figure 7, the design consists of a lower casing, an upper casing, and 10 keys: three circular, two rectangular, and five triangular. The size is due to the low precision and mobility of the child's hands and limbs. The shapes allow the child to recognize the functionality of each key according to its location.

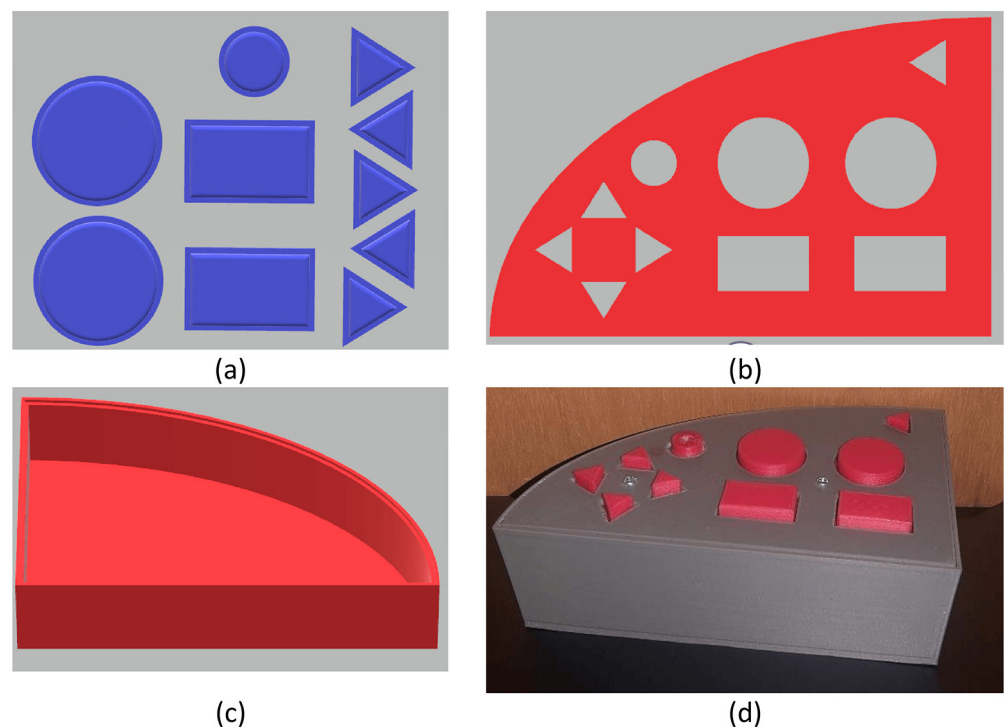


Fig. 7. Modeled and printed adapted keyboard: (a) top view of keys; (b) top view of top cover; (c) isometry of keyboard case; (d) printed keyboard

Connection between matrix keyboard and Arduino. The adapted keyboard consists of a matrix of pushbuttons arranged in columns and rows, and each push-button is connected to a row and a column. This arrangement is known as a matrix keyboard; it is a device that groups several pushbuttons and allows control of them using a lower number of conductors than would be needed when using them individually. In the present development, a 4×3 matrix keyboard is generated, whose pins are connected to the digital input of the Arduino UNO microcontroller. The keyboard uses Arduino UNO and Java for information exchange or communication between the user and the computer [47]. Figure 8 shows the diagram of the connection of the keyboard to the Arduino UNO. Additionally, a screen is provided that

¹ <https://www.shapr3d.com/>

facilitates the simulation of its operation by pressing each of the keyboard's keys and displaying the result.

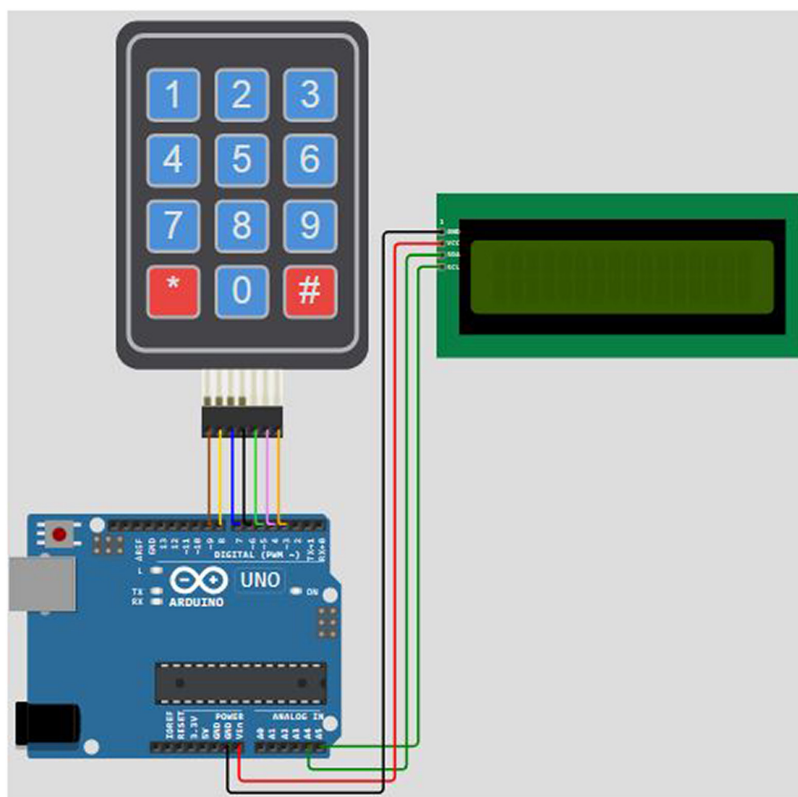


Fig. 8. Micro controlled keyboard system: Wiring diagram and schematic

Key press communication. Matrix keyboards require the user to insert data by pressing keys to write. Therefore, we used the Arduino UNO microcontroller, which allowed us to identify which keys the user presses and determine whether or not the electromagnetic lock is released [48]. The Arduino IDE² and the Keypad.h library³ were used to program the Arduino. The display of digits or letters on our computer screens depends on which key was pressed on the keypad. Once the key is pressed, the microcontroller system identifies which key was closed and proceeds through the serial port (USB) to send a character.

The Arduino UNO communicates with the computer through the serial port. In Java, the RXTX library⁴ is used to communicate with the serial port where the Arduino is connected. The Arduino UNO sends data through the serial port, and the Java program listens to the serial port, reads the data, and performs the keyboard actions in the software according to the strings received. For example, in the writing interface, the child can use the “Scroll Arrows” on the keyboard to scroll through the characters of a word or between words if it is on another line; with the “Escape” key, the child can cancel the process and restart it; with the “Space” key, the child can use spaces between words; with the “Enter” key, the child can accept a word; with the “Character” key, the child can type a letter; with the “Select Character” key, the child can select a word that is auto-completed, and with the “Erase” key, the child can

² <https://www.arduino.cc/en/software>

³ <https://docs.arduino.cc/libraries/keypad/>

⁴ <https://github.com/rxtx/rxtx>

erase the characters of a word if it is misspelled. It is essential to mention that the adapted keyboard is only configured for the “writing interface” of the software. The teacher must manipulate the rest of the interfaces using a normal keyboard.

4.4 Activity

The testing of the functional and nonfunctional requirements allowed corrections to be made at the beginning of the construction of the adapted keyboard and the software. In addition, a series of hardware and software tests were performed using a set of parameters. Table 1 presents the test parameters for the adapted keyboard: durability testing, sensitivity testing, ergonomic testing, and connectivity testing.

Table 1. Testing parameters of the adapted keyboard

Testing	Objective	Method
Durability Testing (HT1)	Ensure that the keyboard can withstand daily use without failure.	Test the keyboard for extended periods, applying different levels of pressure to the keys.
Sensitivity Testing (HT2)	Verify that the keys respond adequately to the pressure applied.	Test with different levels of force to ensure that the keyboard registers keystrokes correctly.
Ergonomics Testing (HT3)	Ensure that the keyboard design is ergonomic for the child.	Test the ergonomics of the keyboard with the child through extended sessions.
Connectivity Testing (HT4)	Ensure that the keyboard is properly connected to the computer.	Test the connection of the keyboard with the computer and different operating systems.

Note: ***HT**: Hardware Testing.

A set of parameters was also proposed to test the software. Table 2 presents the software testing parameters: usability testing, accessibility testing, functionality testing, and integration testing.

Table 2. Testing the parameters of the software

Testing	Objective	Method
Usability Testing (ST1)	Ensure that the software is easy for the child to use.	Test with the child the usability of the software through their interaction.
Accessibility Testing (ST2)	Verify that the software is accessible to the child with multiple disabilities.	Test the data entry options in the writing interface and the use of JAWS.
Functionality Testing (ST3)	Ensure that all software interfaces operate correctly.	Perform extensive testing of each software interface to ensure that they function as intended.
Integration Testing (ST4)	Ensure that software and hardware work well together.	Test the keyboard with the software to ensure that inputs are recorded correctly and evaluate the user experience.

Note: ***ST**: Software Testing.

A scale of 1 to 3 is defined to evaluate the test parameters of the adapted keyboard and software. Table 3 shows the compliance, the rating scale, and the description of

each scale. This scale is intended to evaluate repeatedly the test parameters defined in Tables 2 and 3 until full compliance is achieved.

Table 3. Scale from 1 to 3 to evaluate the test parameters of the adapted keyboard and software

Scale	Value	Description
Full compliance	3	Meets the objective of the testing parameters.
Partial compliance	2	Partially meets the objective of the testing parameters.
Noncompliance	1	Does not meet the objective of the testing parameters.

Different tests and experiments were carried out with the child who recognizes and uses the adapted keyboard. This familiarization stage was critical as it helped to improve both software and hardware aspects. During the testing, it was important to have the help of the center's parents and caregivers, who could make important comments to both the user and the developers.

The user tests the adapted keyboard and software for learning to write using a computer. It should be noted that in the first week of testing, partial compliance was obtained in the HT3 and ST3 test parameters. In addition, non-compliance was obtained for test parameters HT4 and ST4. In the second week of testing, there was still partial compliance in HT4, ST3, and ST4 test parameters. Finally, full compliance was achieved for the test parameters in the third week. An improvement was noted in the student's writing using the adapted keyboard, software, and computer. However, the teacher and the student's family should enter new words into the software to progress with writing according to the new learning needs.

5 DISCUSSION

Current assistive technologies have limitations, such as high costs and poor ergonomic designs [49] that reduce patient adherence to use. The WHO Global Report on Assistive Technology [50] highlighted global inequalities in access to assistive products, reinforcing the value of low-cost, user-centered tools such as those proposed in this work. The design of the adapted keyboard for a child with motor impairment was derived from anthropological measurements that considered ergonomics in a seated position. Therefore, the angular data of the joint of the child's upper limb with limited movement were measured. This study was necessary to delimit the range of action of the upper limbs on the keyboard. In addition, the trunk's posture was considered, as this can hinder ergonomic posture at the computer. The number of keys and their arrangement depended on the individual's reach and motor skills. Also, we have considered human-computer interaction (HCI), which is a field of computer science that studies the interaction between people (users) and computers, considering a wide range of issues, such as user interface design, human factors, and social and psychological aspects of technology use. Therefore, the design, implementation, and evaluation of user interfaces are the focal points of HCI [51]. HCI researchers aim to make computer systems more accessible and usable. HCI is multidisciplinary, using computer science, psychology, and design to develop technologies that meet the needs of users [52].

Consequently, other authors [53] highlighted the importance of implementing assistive technologies for students with cerebral palsy through participatory and teacher-informed approaches. In addition, other authors [54] have explored

the integration of educational robotics in special education, providing insights into inclusive learning through accessible technologies. Similarly, other authors [55] have examined the combined use of virtual reality and clinical hypnosis in special education settings, supporting digital tools to improve engagement and cognitive-emotional regulation in students with disabilities. In summary, assistive technologies encourage the educational inclusion of people with multiple disabilities to improve their quality of life using various technologies.

This study's limitations include a single participant, a single-case design, a lack of quantitative outcome measures, and short-term evaluation, which limit evidence and results. In addition, challenges related to scalability, maintenance, or adaptation to broader populations are not addressed.

6 CONCLUSIONS AND FUTURE WORK

The objective of this study was to design and implement an adapted keyboard for a child with spastic tetraplegia, visual and motor disabilities. In addition, software was created for the child to interact with the adapted keyboard and the computer. The adapted keyboard was designed considering the ergonomics of the child's sitting position and the range of motion of his upper limbs. Given that the study was conducted in an LMIC, it was imperative that the solution developed was low-cost.

The adapted keyboard and software for learning to write on a computer are assistive technologies. They aim to improve the quality of life of the child under study and aid in educational inclusion. The continuous evolution of assistive technologies in hardware and software solutions makes it possible to meet the needs of people with disabilities according to their conditions and environment. The main result is the creation of the adapted keyboard using an Arduino UNO microcontroller. The software for learning to write is supported by the layered architecture. In addition, the software allows interaction with the JAWS screen reader, as it has been developed using accessibility principles. The software allows auto-completion of words. In addition, it allows us to select options based on the child's vocabulary and reduce writing time.

In future work, word auto-completion will be developed using artificial intelligence tools to learn to write on a computer. Integrating artificial intelligence into the software could help prioritize word choice through auto-completion based on the individual's vocabulary, thus reducing typing time. Otherwise, the number of keys should be reduced, and their size increased. In other future work, the number and size of keys will be studied to determine whether a more optimal or general solution useful to the greatest number of users can be achieved. In other future work, we propose to evaluate the adapted software and keyboard with quantitative data measuring typing speed, accuracy, and user satisfaction. We will also integrate other relevant theories of rehabilitation, special education, and human-computer interaction. We will also address psychosocial aspects (e.g., self-efficacy, social inclusion, family and caregiver perspectives) that are crucial for technology adoption and use.

7 ACKNOWLEDGMENTS

This work was supported by the Software Engineering Center and RobLab of the Catholic University of Cuenca as part of the scientific research project PICCG24-16, providing the necessary equipment to develop it.

8 REFERENCES

- [1] M. Jordan, G. N. Nogueira, A. Brito, and P. Nohama, "Virtual keyboard with the prediction of words for children with cerebral palsy," *Computer Methods and Programs in Biomedicine*, vol. 192, p. 105402, 2020. <https://doi.org/10.1016/j.cmpb.2020.105402>
- [2] M. Hersh and M. Johnson, "On modelling assistive technology systems – Part I: Modelling framework," *Technology and Disability*, vol. 20, no. 3, pp. 193–215, 2008. <https://doi.org/10.3233/TAD-2008-20303>
- [3] R. Islam, S. Rahman, and A. Sarkar, "Computer vision based eye gaze controlled virtual keyboard for people with quadriplegia," in *2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI)*, 2021, pp. 1–6. <https://doi.org/10.1109/ACMI53878.2021.9528213>
- [4] J. G. Enríquez, L. M. Soria Morillo, J. A. García-García, and J. A. Álvarez-García, "Two decades of assistive technologies to empower people with disability: A systematic mapping study," *Disability and Rehabilitation: Assistive Technology*, vol. 19, no. 5, pp. 2095–2112, 2024. <https://doi.org/10.1080/17483107.2023.2263504>
- [5] S. Jeon, H. Lee, J. Jung, and J. R. Kim, "User-adaptive key click vibration on virtual keyboard," *Mobile Information Systems*, vol. 2018, no. 1, 2018. <https://doi.org/10.1155/2018/6126140>
- [6] T. Alquraini and D. Gut, "Critical components of successful inclusion of students with severe disabilities: Literature review," *International Journal of Special Education*, vol. 27, no. 1, pp. 42–59, 2012.
- [7] Y. Gu, C. Yu, and Y. Shi, "The dynamic grouping keyboard: A general keyboard optimization approach for users with motor impairment," *CCF Transactions on Pervasive Computing and Interaction*, vol. 1, pp. 89–99, 2019. <https://doi.org/10.1007/s42486-019-00010-5>
- [8] C. Chiapparino, F. Stasolla, C. de Pace, and C. Lancioni, "A touch pad and a scanning keyboard emulator to facilitate writing by a woman with extensive motor disability," *Life Span and Disability*, vol. 14, no. 1, pp. 45–54, 2011.
- [9] F. Heilmann, "Executive functions and domain-specific cognitive skills in climbers," *Brain Sciences*, vol. 11, no. 4, pp. 1–11, 2021. <https://doi.org/10.3390/brainsci11040449>
- [10] Y. Lagarrigue, C. Cappe, and J. Tallet, "Regular rhythmic and audio-visual stimulations enhance procedural learning of a perceptual-motor sequence in healthy adults: A pilot study," *PLoS ONE*, vol. 16, no. 11, pp. 1–21, 2021. <https://doi.org/10.1371/journal.pone.0259081>
- [11] M. V. Silveira, M. Silvestrin, E. C. Vilela, J. C. de Rose, E. Arntzen, and M. S. Caetano, "Equivalence relations do exist before they are tested: Confirmatory evidence revealed by EEG measurements," *Journal of the Experimental Analysis of Behavior*, vol. 115, no. 1, pp. 284–295, 2021. <https://doi.org/10.1002/jeab.662>
- [12] S. Zhu, T. Luo, X. Bi, and S. Zhai, "Typing on an invisible keyboard," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*, 2018, pp. 1–13. <https://doi.org/10.1145/3173574.3174013>
- [13] I. C. Rodrigues Almeida, J. Ribeiro, and A. Moreira, "Assistive technologies for children with cognitive and/or motor disabilities: Interviews as a means to diagnose the training needs of informal caregivers," *Disability and Rehabilitation: Assistive Technology*, vol. 16, no. 3, pp. 340–349, 2019. <https://doi.org/10.1080/17483107.2019.1680750>
- [14] M. Visser *et al.*, "The use of assistive technology in classroom activities for learners with motor impairments at a special school in South Africa," *South African Journal of Occupational Therapy*, vol. 50, no. 2, pp. 11–22, 2020. <https://doi.org/10.17159/2310-3833/2020/vol50no2a3>

- [15] L. Oad, "Journey of inclusive education exploring its evolution and future perspectives," *Spry Journal of Literature and Linguistics*, vol. 1, no. 1, pp. 42–58, 2023. <https://doi.org/10.62681/sprypublishers.sjll/1/1/4>
- [16] M. Onyesom and C. O. Igberaharha, "Inclusive values and pedagogies needed by business studies teachers for effective inclusive education in secondary schools," *International Journal of Education and Practice*, vol. 9, no. 1, pp. 220–229, 2021. <https://doi.org/10.18488/journal.61.2021.91.220.229>
- [17] T. Makoele, "Inclusive education: Are we there? some global challenges, contradictions and anomalies," *Journal of Sociology and Social Anthropology*, vol. 5, no. 3, pp. 303–309, 2014. <https://doi.org/10.31901/24566764.2014/05.03.04>
- [18] E.-J. Hoogerwerf, K. Mavrou, and I. Traina, Eds., *The Role of Assistive Technology in Fostering Inclusive Education: Strategies and Tools to Support Change*. London: Routledge, 2020. <https://doi.org/10.4324/9780429428241>
- [19] G. Esposito, P. B. Marschik, A. Nordahl-Hansen, and A. Brandelli Costa, "Developmental disabilities in low resource contexts," *Research in Developmental Disabilities*, vol. 140, p. 104589, 2023. <https://doi.org/10.1016/j.ridd.2023.104589>
- [20] A. M. Cook and J. M. Polgar, *Assistive Technologies: Principles and Practice* (4th ed.). St. Louis, MO: Mosby, 2015.
- [21] A. Z. Hameed and A. M. Basahel, "Investigation of work-related disorders by rapid upper limb assessment," *Journal of Scientific & Industrial Research*, vol. 78, pp. 199–202, 2019. <https://nopr.niscpr.res.in/bitstream/123456789/46942/1/JSIR%2078%284%29%20199-202.pdf>
- [22] A. R. Mohd Nasrull *et al.*, "Relationship between improper working posture among maintenance workers using different assessment methods," *Journal of Advanced Research in Applied Sciences and Engineering Technology*, vol. 49, no. 2, pp. 15–25, 2025. <https://doi.org/10.37934/araset.49.2.1525>
- [23] R. Kasauli, E. Knauss, J. Horkoff, G. Liebel, and F. Gomes de Oliveira Neto, "Requirements engineering challenges and practices in large-scale agile system development," *Journal of Systems and Software*, vol. 172, pp. 1–26, 2021. <https://doi.org/10.1016/j.jss.2020.110851>
- [24] P. Macedo, R. N. Madeira, P. Miranda, and P. A. Santos, "Crafting personalised web interfaces: Enhancing accessibility for persons with disabilities," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 20, no. 13, pp. 84–102, 2024. <https://doi.org/10.3991/ijoe.v20i13.50025>
- [25] F. Stasolla, A. O. Caffò, L. Picucci, and A. Bosco, "Assistive technology for promoting choice behaviors in three children with cerebral palsy and severe communication impairments," *Research in Developmental Disabilities*, vol. 34, no. 9, pp. 2694–2700, 2013. <https://doi.org/10.1016/j.ridd.2013.05.029>
- [26] S. Pouplin *et al.*, "Effect of dynamic keyboard and word-prediction systems on text input speed in persons with functional tetraplegia," *The Journal of Rehabilitation Research and Development*, vol. 51, no. 3, pp. 467–480, 2014. <https://pubmed.ncbi.nlm.nih.gov/25019668/>
- [27] C. L. Mitchell *et al.*, "Ability-based methods for personalized keyboard generation," *Multimodal Technologies and Interaction*, vol. 6, no. 8, pp. 1–20, 2022. <https://doi.org/10.3390/mti6080067>
- [28] T. Marinaci *et al.*, "An inclusive workplace approach to disability through assistive technologies: A systematic review and thematic analysis of the literature," *Societies*, vol. 13, no. 11, pp. 1–23, 2023. <https://doi.org/10.3390/soc13110231>
- [29] F. N. Lee, H. A. Feldner, K. Hsieh, F. Balcazar, B. S. Bonfiglio, and S. Parker, "The interdependence-Human Activity Assistive Technology Model (i-HAAT): A contemporary conceptual approach to assistive technology outcomes research," in *Proceedings of the RESNA Annual Conference 2022*, Washington, 2022, pp. 1–4.

- [30] S. L. Clay and R. Alston, "Assistive technology use and veterans: An examination of racial differences between whites and blacks using the HAAT model," *Journal of Vocational Rehabilitation*, vol. 45, no. 2, pp. 159–171, 2016. <https://doi.org/10.3233/JVR-160820>
- [31] United Nations, "Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all." [Online]. Available: <https://sdgs.un.org/goals/goal4>
- [32] UNESCO, "Information ethics," 2021. [Online]. Available: <https://www.unesco.org/en/ifap/information-ethics>
- [33] Global Education Monitoring Report Team, "Learners with disabilities and technology: Advocacy brief," 2024. [Online]. Available: <https://unesdoc.unesco.org/ark:/48223/pf0000389161>
- [34] P. F. Hunt, "Inclusive education: The case for early identification and early intervention in assistive technology," *Assistive Technology*, vol. 33, no. sup1, pp. S94–S101, 2021. <https://doi.org/10.1080/10400435.2021.1974122>
- [35] H. Koester, R. Simpson, and J. Mankowski, "Software wizards to adjust keyboard and mouse settings for people with physical impairments," *The Journal of Spinal Cord Medicine*, vol. 36, no. 4, pp. 300–312, 2013. <https://doi.org/10.1179/2045772312Y.0000000049>
- [36] Z. Hoy and M. Xu, "Agile software requirements engineering challenges-solutions—A conceptual framework from systematic literature review," *Information*, vol. 14, no. 6, pp. 1–19, 2023. <https://doi.org/10.3390/info14060322>
- [37] P. Kumar, A. Khan, A. Ahmad, K. Khalid, and M. Wasim, "Software engineering requirement model," *International Journal of Scientific & Engineering Research*, vol. 9, no. 11, pp. 523–529, 2018.
- [38] S. Tiun, U. Mokhtar, S. Bakar, and S. Saad, "Classification of functional and non-functional requirement in software requirement using Word2vec and fast Text," *Journal of Physics: Conference Series*, vol. 1529, pp. 1–6, 2020. <https://doi.org/10.1088/1742-6596/1529/4/042077>
- [39] L. Chung and J. C. do Prado Leite, "On non-functional requirements in software engineering," in *Conceptual Modeling: Foundations and Applications*, in Lecture Notes in Computer Science, A. T. Borgida, V. K. Chaudhri, P. Giorgini, and E. S. Yu, Eds., vol. 5600, 2009, pp. 363–379. https://doi.org/10.1007/978-3-642-02463-4_19
- [40] R. Kumar *et al.*, "Development of a prototype global positioning system based stick for blind patients," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 20, no. 8, pp. 21–36, 2024. <https://doi.org/10.3991/ijoe.v20i08.49343>
- [41] B. Bruegge and A. H. Dutoit, *Object-Oriented Software Engineering: Using UML Patterns and Java* (3rd Edition). Prentice Hall: Pearson, 2010.
- [42] World Wide Web Consortium, "Providing accessible names and descriptions," 2024. [Online]. Available: <https://www.w3.org/WAI/ARIA/apg/practices/names-and-descriptions/>
- [43] S. Harper, G. Khan, and R. Stevens, "Design checks for java accessibility," in *Accessible Design in the Digital World Conference*, 2005, pp. 1–8. <https://doi.org/10.14236/ewic/AD2005.11>
- [44] S. Saleh, M. Rahman, and K. Pavel, "Comparative study on the software methodologies for effective software development," *International Journal of Scientific and Engineering Research*, vol. 8, no. 4, pp. 1018–1025, 2017.
- [45] R. Scott, "Example: Adding autocomplete to JTextField," Stack Abuse, 2023. [Online]. Available: <https://stackabuse.com/example-adding-autocomplete-to-jtextfield/>
- [46] I. Sabry, T. El-Attar, and A. M. Hewidy, "Optimization of fused deposition modelling acrylonitrile-co-butadiene-co-styrene parameters using ANOVA and Hybrid GRA-TOPSIS," *Journal of Advanced Research in Applied Sciences and Engineering Technology*, vol. 50, no. 1, pp. 66–77, 2025. <https://doi.org/10.37934/araset.50.1.6677>

- [47] S. Shreevidya, N. Namratha, V. M. Nisha, and M. Dakshayini, "Hand gesture based human-computer interaction using arduino," in *EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing*, in EAI/Springer Innovations in Communication and Computing, A. Haldorai, A. Ramu, S. Mohanram, and C. Onn, Eds., 2020, pp. 315–321. https://doi.org/10.1007/978-3-030-19562-5_31
- [48] A. Afroz, "Digital smart door lock security system using arduino uno microcontroller," *Iconic Research and Engineering Journals*, vol. 6, no. 1, pp. 153–159, 2022.
- [49] B. Luque-Milera, W. K. Sánchez-Contreras, E. J. L. Florian-Amaro, E. L. Acosta-Valdez, J. M. Sanchez-Ramirez, and C. Castro-Vargas, "Design and implementation of a force sensor glove for post-accident patient rehabilitation," *International Journal of Online and Biomedical Engineering (ijOE)*, vol. 21, no. 7, pp. 88–105, 2025. <https://doi.org/10.3991/ijoe.v21i07.54273>
- [50] World Health Organization, "Global report on assistive technology," 2022. [Online]. Available: <https://www.who.int/publications/i/item/9789240049451>
- [51] P. Brey and J. H. Søraker, "Philosophy of computing and information technology," in *Philosophy of Technology and Engineering Sciences*, Anthonie Meijers, Ed., 2009, pp. 1341–1407. <https://doi.org/10.1016/B978-0-444-51667-1.50051-3>
- [52] S. AlZu'bi, A. Mughaid, F. Quiam, and S. Hendawi, "Exploring the capabilities and limitations of ChatGPT and alternative big language models," *Artificial Intelligence and Applications*, vol. 2, no. 1, pp. 28–37, 2023. <https://doi.org/10.47852/bonviewAIA3202820>
- [53] S. Botha and M. Mihai, "The use of assistive technology to minimize educational learning barriers for learners with cerebral palsy," *Journal of Education*, no. 93, pp. 23–42, 2024. <https://doi.org/10.17159/2520-9868/i93a02>
- [54] E. Chaidi, C. Kefalis, Y. Papagerasimou, and A. Drigas, "Educational robotics in primary education. A case in Greece," *Research, Society and Development*, vol. 10, no. 9, pp. 1–12, 2021. <https://doi.org/10.33448/rsd-v10i9.16371>
- [55] A. Drigas, E. Mitsea, and C. Skianis, "The role of clinical hypnosis & VR in special education," *International Journal of Recent Contributions from Engineering, Science & IT (ijES)*, vol. 9, no. 4, pp. 4–18, 2021. <https://doi.org/10.3991/ijes.v9i4.26147>

9 AUTHORS

Milton Campoverde-Molina received a Ph.D. degree in Information and Communication Technologies from the Department of Mathematics and Computer Science, University of the Balearic Islands, in Spain, in 2022; the Master's degree in Evaluation and Audit of Technological Systems from the University of the Armed Forces – ESPE (Ecuador), in 2015; the Master's degree in University Teaching from the University of the Armed Forces – ESPE (Ecuador), in 2014; and the title of Systems Engineering from the Catholic University of Cuenca (Ecuador), in 2009. He is a tenured Professor at the Academic Unit of Informatics, Computer Science, and Technological Innovation at the Catholic University of Cuenca (Ecuador). In recent years, he has been involved in web accessibility research. He has published chapters in books and articles at several conferences and journals. His main research topics include Web Accessibility, Education, and Software Engineering (E-mail: mcampoverde@ucacue.edu.ec).

Juan-Carlos Cobos-Torres is a researcher at the Catholic University of Cuenca, Postgraduate Sub direction. He received his Ph.D. at Carlos III University of Madrid and received his MSc degree from the same University. His main research of interest focuses on real-time computer vision techniques applicable to robotics or other demanding real-world, and real-time applications. Further, Biomedical Signal

Processing with methods and algorithms based on computer analysis of biological and medical images, in particular, vital sign detection. Finally, his research interests also include e-learning in higher education, gamification learning, problem-based learning, among others (E-mail: juan.cobos@ucacue.edu.ec).

Sergio Luján-Mora received a Ph.D. degree in computer engineering from the Department of Software and Computing Systems, University of Alicante, in Spain, in 2005 and a Computer Science and Engineering degree from the University of Alicante, in 1998. He is currently a Senior Lecturer with the Department of Software and Computing Systems, University of Alicante. In recent years, he has focused on elearning, massive open online courses (MOOCs), open educational resources (OERs), and the accessibility of video games. He is the author of several books, and many published articles in various conferences, including ER, UML, and DOLAP, and high-impact journals, including DKE, JCIS, JDBM, JECR, JIS, JWE, IJEE, and UAIS. His main research interests include web applications and web development, and web accessibility and usability (E-mail: sergio.lujan@ua.es).

Carlos Paúl Ortiz-Alvarez received his degree in Electrical Engineering from Universidad Católica de Cuenca. He currently works in the area of design, implementation, and supervision of electrical systems for residential and commercial installations, with a focus on home automation and advanced automation. Finally, his research interests also include the design process of objects based on modeling and prototyping techniques, learning embedded systems, among others.

Eugenio Esteban Palomeque-Zambrano is a Software Engineer, graduated in 2023 from the Catholic University of Cuenca. He currently works as a software and web applications developer, with a special focus on serverless technologies and the use of artificial intelligence models LLM. His practical experience is complemented with the contribution and development of accessible web applications.