ProtoBricks: A Research Toolkit for Tangible Prototyping & Data Physicalization

Citation for the original published paper (version of record):
Dominiak, J., Walczak, A., Stefanidi, E. et al (2024). ProtoBricks: A Research Toolkit for Tangible Prototyping & Data Physicalization. Proceedings of the 2024 ACM Designing Interactive Systems Conference, DIS 2024: 476-495. http://dx.doi.org/10.1145/3643834.3661573

ProtoBricks: A Research Toolkit for Tangible Prototyping & Data Physicalization

Julia Dominiak
julia.dominiak@p.lodz.pl
Lodz University of Technology
Łódź, Poland

Anna Walczak
anna.walczak@ubicomp.pl
Lodz University of Technology
Łódź, Poland

Evropi Stefanidi
evropi@uni-bremen.de
University of Bremen
Bremen, Germany

Krzysztof Adamkiewicz
kadamkiewicz835@gmail.com
Lodz University of Technology
Łódź, Poland

Krzysztof Grudzień
kgrudzi@iis.p.lodz.pl
Lodz University of Technology
Łódź, Poland

Jasmin Niess
jasminni@uio.no
University of Oslo
Oslo, Norway

Paweł W. Woźniak
pawel.wozniak@tuwien.ac.at
Chalmers University of Technology
Gothenburg, Sweden
TU Wien
Vienna, Austria

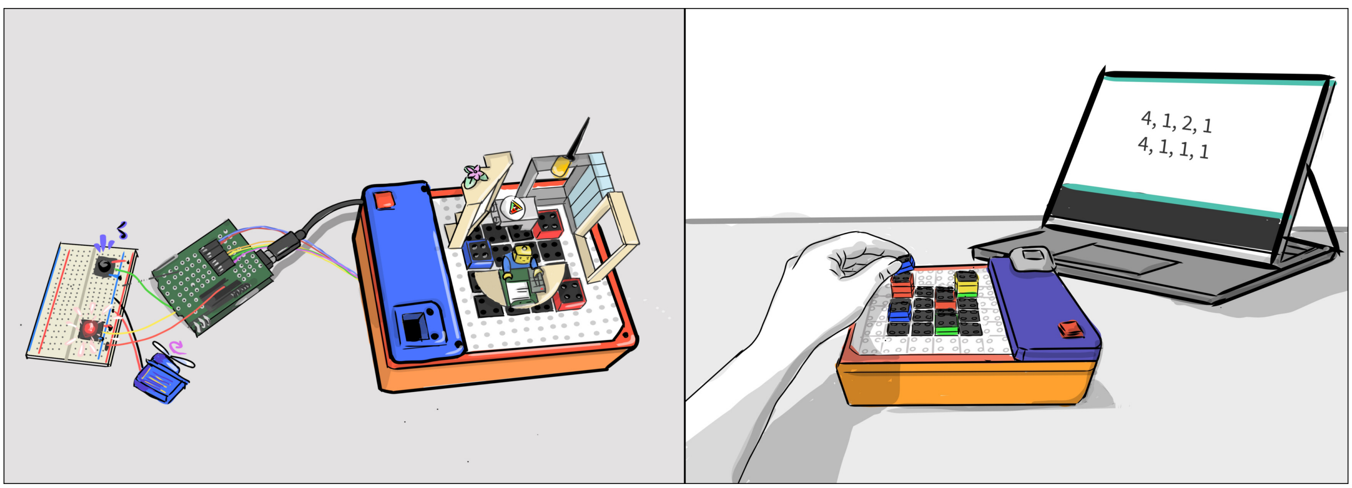

Figure 1: ProtoBricks is a versatile and replicable system for Tangible User Interacfes prototyping and data physicalization.

ABSTRACT

Building tangible interfaces or data physicalizations is a resource-intensive endeavour. There is a need for rapid means to prototype tangibles in order to facilitate research and design. To this end, we designed ProtoBricks: a research toolkit that uses capacitive bricks to facilitate rapid prototyping for tangible interfaces. Utilizing toy bricks that do not contain electronics, ProtoBricks can record brick position and color. Specialized knowledge is not required to build our system as it uses widely available components and 3D printing. We contribute the full software and hardware specification of the toolkit. We evaluate the utility of the toolkit by reporting on past use cases and prototyping workshops. We show that the toolkit facilitates creativity and effectively supports prototyping. ProtoBricks lowers the entry threshold for experimenting with tangible interfaces and enables researchers and designers to focus on the interaction with their prototype, delegating implementation to the toolkit.

DIS '24, July 01–05, 2024, IT University of Copenhagen, Denmark

ACM ISBN 979-8-4007-0583-0/24/07
https://doi.org/10.1145/3643834.3661573

CCS CONCEPTS

• **Human-centered computing** → **User interface toolkits**.

KEYWORDS

tangible, prototyping/implementation, artefact or system

ACM Reference Format:

Julia Dominiak, Anna Walczak, Evropi Stefanidi, Krzysztof Adamkiewicz, Krzysztof Grudzień, Jasmin Niess, and Paweł W. Woźniak. 2024. ProtoBricks: A Research Toolkit for Tangible Prototyping & Data Physicalization. In *Designing Interactive Systems Conference (DIS '24), July 01–05, 2024, IT University of Copenhagen, Denmark.* ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3643834.3661573

1 INTRODUCTION

Toy bricks have been entertaining children and adults for centuries. Over the years, the materials, manufacturing techniques, and shape of these toys have changed, but the idea of interacting with physical bricks in space remains popular among different groups of users around the world [3]. Recent technological advancements brought new capabilities for interacting with bricks, making them more dynamic and open for new applications. Previous works introduced a number of interactive brick systems, exploring both the technical aspects of interlacing bricks with electronics, and the design space of new interaction techniques that they provide [17]. Moreover, the inherent tangibility of bricks hints at their potential as building blocks of Tangible User Interfaces (TUI) [4, 40] or accessible data physicalization tool [8]. However, while previous endeavours in HCI presented different approaches to designing interactive brick systems, they usually required expensive external equipment [67], knowledge about electronics and computer science to operate [35] or were focused on very specific applications [44, 52]. Furthermore, although HCI developed a rich body of work on data physicalization and its benefits for understanding and reflecting on the data [28], we lack a versatile solution facilitating the physicalization of data in different contexts for research and personal purposes. To address these issues, we developed the ProtoBricks toolkit—an interactive brick system for data physicalization and tangible interface prototyping (see Figure 1).

ProtoBricks can be built using 3D printing with simple electronics. The main parts are dedicated bricks and a base plate, which were printed using a conductive, composite PLA filament. The system relies on detecting changes in capacity through sensors hidden in the base. Recording activities involving bricks and interacting with the structure in 3D space is captured by thresholding the read value, derived from the user's hand contact via a conductive filament. The open architecture of both the hardware and Arduino microcontroller-based software enables versatile application and adaptation to user needs. Toolkit enables using bricks as digital input signals, enabling building versatile tangible interfaces. Moreover, our toolkit facilitates the recording of activity logs, which can contribute to easier measurement during data physicalization.

In our work, we draw on the natural properties of toy bricks to aid researchers with an accessible, versatile tool, employing affordances of well-known toys played by people in different age groups and cultures. Moreover, we developed ProtoBricks to facilitate measurement and allow for further extensions and modifications. We designed ProtoBricks to address the needs of two primary audiences: researchers and educators. For researchers, the toolkit can serve as a versatile research probe, possible to tailor to the specific study scenario or employed as a tangible prototype suitable for diverse participant groups. For educators, ProtoBricks can act as a resource to teach electronics, prototyping, and Arduino programming. We evaluated our toolkit using several evaluation strategies, following recommendations by Ledo et al. [37]. First, in two case studies, we demonstrated how ProtoBricks can be used as a research probe in studies exploring tangible user interfaces. Then, we investigated the usage of our toolkit in a series of participatory workshops, gathering insights from individuals with varying experience in prototyping and electronics. Finally, in a series of expert interviews with educators, we explored ProtoBricks' capability to act as educational tool.

This paper contributes (1) the design and implementation of an interactive brick toolkit for data physicalization and Tangible User Interface prototyping, (2) detailed instructions for replicating and using the toolkit, (3) insights from an empirical evaluation of the ProtoBricks toolkit, including demonstrations of using it in user-studies, and (4) directions and tools for further extensions and modifications.

2 RELATED WORK

In this section, we present prior research that inspired and informed the design of ProtoBricks toolkit, covering (1) different technical approaches to detect and interact with bricks in 3D space, (2) Tangible User Interfaces, and the benefits of data physicalization, and (3) the use of toolkits in HCI.

2.1 Tracking and interacting with bricks in space

Previous research explored various technical approaches to using bricks as tangible controllers [52], primarily for digitizing physical collections into databases [50, 60, 70] or proposing builds [70], with the aim of enhancing interactions. Many of these systems were based on RGB or RGB-D camera image analysis [20, 52], most often supported by neural networks (machine learning) [50, 60, 70] for bricks detection and tracking. While there exist a number of image-recognition algorithms, introducing new colors or shapes of bricks requires modifying the program, making it complex and hard to follow [20, 52]. Employing machine learning helps to overcome these limitations but introduces new challenges, e.g., time-consuming training of the network and usually advanced hardware [60]. Nevertheless, the implementation of brick recognition based on a vision system has been used in commercial solutions. Rebrickable® [50] developed a mobile application providing a comprehensive list of bricks after scanning the image of owned LEGO® bricks. Another app—Brickit [70] suggests possible builds with LEGO® bricks based on a photo of available parts. While these examples showcase the capabilities of vision systems to brick detection, such an approach requires external apparatus that has to be placed above the work surface or model, which can considerably impede the system's versatility and necessitates capturing the shot from an overhead perspective. While vision systems are powerful tools for object detection, they come with a number of limitations, which strip the toy bricks from their ease of use and availability. Therefore, HCI lacks the tools to allow for tangible interaction without the need for complex software solutions or expensive external devices.

Prior studies in Human-Computer Interaction (HCI) has investigated how interacting with bricks on a tabletop facilitates real-time analysis of the shape and position of the construction's base. This analysis can be achieved through built-in cameras [13] or infrared sensors [6, 44], and is often enhanced by the use of tangible object applications such as game pieces and controllers [19, 59]. Furthermore, these systems support the physicalization of data through brick representations like notes [44] or content from white-board [19, 34]. They also track objects by register logs and critique constructions, which can be implemented in games or used to digitally encapsulate information tangibly [6]. Moreover, both tabletops and smaller screens (tablets/smartphones) based on capacitance measurement allow the use of RFID tags, capacitors, or conductive 3D printing as a trigger for detecting bricks [19, 34, 59]. While tabletops provide a variety of ways to detect and track the bricks, they are usually limited to the analysis of the base of the build. To overcome this limitation, Chan et al. [13] introduced the capacitive technique of analyzing the blurring of a marker image. However, similarly to vision systems, these solutions require expensive external screens to operate on, limiting the interaction to a dedicated workstation. In our work, we developed a system for bricks-play that remains mobile, allowing for playing in various environments.

Another approach to implementing interactive bricks includes embedding the active or passive electronics inside. This implementation primarily focuses on the technical preparation of elements, tracking changes position of physical elements in 3D space, and visualizing these changes in the GUI through the presentation of the construction [25, 26, 69]. Rapidly developing 3D printing techniques allow for designing customized shapes and sizes using classic materials and flexible or conductive filaments. Moreover, 3D-printing facilitates replication [10]. Using 3D printing, Yoshida et al. [69] developed Capacitive Blocks—bricks acting as capacitors. This idea was later extended by Ikegawa et al. [26], who introduced Lightweight Capacitance—bricks with electronics hidden inside, which increased the detection accuracy. However, this approach limited the detection of higher construction, as stacking more than four bricks notably impeded the detection. Therefore, Ikegawa et al. [25] introduced Tesla Blocks, where magnets were inserted into the popular Duplo bricks, allowing for precise detection of higher structures. Moreover, RFIBricks [23] or NFCStack [39] operate without the need for an expensive display base, enabling the construction of interactive 3D geometry using passive stackable blocks. However, although electronic components miniaturize rapidly [66], it is still difficult to embed them into toy bricks while maintaining the size and shape of the traditional bricks. Despite the advancements made in the field of interactive brick systems, HCI still lacks an easy and accessible solution for designing tangible bricks-based interfaces, allowing for tracking bricks detection in 3D space and preserving the affordances of popular toy bricks.

2.2 Physicalization and tangible user interfaces

Representing abstract information in a physical form has been used from prehistory [29, 55]. Throughout history, people developed a number of ways to translate numbers, ideas, and messages into physical forms, including mock-ups, sculptures, models, or installations [28]. With the growing popularity of LEGO®, toy bricks have become a popular material for physicalizing spatial planning [21, 65], personal health data [9] or even a calendar [58]. One of the key characteristics of toy bricks facilitating physical data representations is the large variety of available shapes and colors and the possibility to re-use and dynamically manipulate. Moreover, toy bricks allow for stable connections while also remaining modular and re-shapable of construction, which positively influences the satisfaction of the interaction [24, 49]. However, while toy bricks are a powerful tool for physicalizing abstract concepts, traditional bricks do not allow for the dynamic link between digital data and their tangible representation. While prior research proved data physicalization to foster reflection, creativity, and understanding of complex phenomena, flat, paper- or screen-based representations have been prevailing due to their versatility and availability [28]. In contrast, Tangible User Interfaces (TUIs) allow for interacting with physical objects associated with digital information, connecting the physical and digital worlds. HCI developed a rich body of work on Tangible User Interfaces, covering a number of applications, including education [15], health [14], collaboration [1] and leisure activities [41]. However, implementing high-fidelity TUIs requires electronics knowledge, time to design, implement, and test the solutions, and often advanced hardware resources, making Tangible User Interfaces less accessible for some researchers. Therefore, HCI lacks versatile, replicable, and easy-to-use tools for rapidly implementing Tangible User Interfaces for research and educational purposes. In our work, we aim to bridge this gap by developing a reconfigurable, reusable TUI, the implementation of which requires minimal knowledge of electronics and limited hardware resources.

2.3 Toolkits in HCI

Developing prototypes for new systems is a core activity in HCI. These prototypes serve not only as a medium for exploration, testing, and iteration but also as an effective solution to challenges in user interfaces. Nonetheless, prototype development is not always the primary research focus. There are instances when the investment in prototype development does not align with the results achieved. For such scenarios, toolkits serve as intermediate research artifacts, offering insights into creating interactive systems [37]. This allows users (researchers) to employ them without complete knowledge of how the solution works from a technical point of view, as delineated by Oulasvirta et al. [45]. Ledo et al. [37] outline the objectives of toolkit development: reducing authoring time and complexity, creating paths of least resistance, empowering new audiences, integrating with existing practices and infrastructures, and enabling replication and creative exploration. Our work uses these goals as guiding principles for developing ProtoBricks.

Transforming physical objects and interfaces into digital forms through tangibles has been widely explored, particularly in terms of universal and flexible surfaces. Examples of such sensor systems include Project Zanzibar [63], the VoodooIO Gaming Kit [64], and VoodooIO [62] itself. These systems serve as sensor platforms that seamlessly integrate tangible interactions with digital environments. Each system showcases how flexible and adaptable surfaces can be used not only to detect and interact with physical objects but also to enhance user control and customization. This merger of physical inputs with digital outputs facilitates enriched interactive

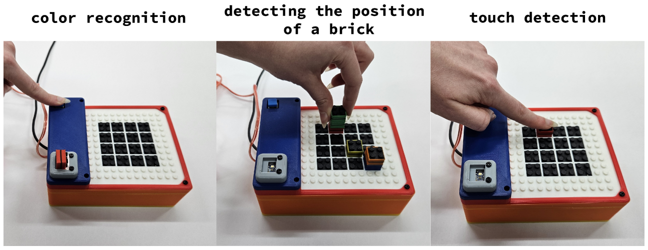

Figure 2: The basic features of ProtoBricks are scanning and color recognition, recognizing and recording the addition or removal of a brick, and detecting and locating the position of touching the brick.

experiences. Such projects play a crucial role in advancing TUI technology and in making physicalization easy and accessible to implement. However, these projects focused solely on the detection and integration with objects in two-dimensional space.

Our work is also inspired by past efforts in toolkits designed specifically for tangible interfaces. Villanueva et al. introduced *ColabAR*, a toolkit designed for remote collaboration in Tangible Augmented Reality (TAR) laboratories, leveraging physical proxies to manipulate virtual objects [61]. Steuerlein and Mayer developed a toolkit for everyone to train deep learning tangible recognizers, mainly focusing on capacitive screens [57]. Petrelli et al. explored how heritage installations can be supported by a tangible interaction toolkit [47]. Earlier, Klemmer et al. introduced *Papier-Mâché*, which supports tangible input through a combination of computer vision, electronic tags, and barcodes, demonstrating the breadth of toolkit possibilities in this domain [33]. Our work is interestingly different from these toolkits as it centers on building tangibles from bricks, enabling its users to create complex interfaces from discrete parts.

Historical research in the HCI domain underscores the value of evaluating toolkits. However, it also recognizes the inherent challenges in conducting toolkit evaluations [37, 43]. Four evaluation strategies for HCI toolkits have been proposed by Ledo et al. [37]: Demonstration, Usage, Technical Performance, and Heuristics. In our study, we adopt two of these strategies: Demonstration and Usage to verify and evaluate the requirements made for the ProtoBricks toolkit: (1) *Does our toolkit facilitates data physicalization and rapid prototyping of TUIs?*, (2) *Is the ProtoBricks toolkit suitable for using in both research and educational settings?* and (3) *Does the toolkit support creativity during usage?*

3 THE PROTOBRICKS TOOLKIT

We developed ProtoBricks to facilitate two main purposes: data physicalization and TUI prototyping. ProtoBricks addresses the needs of two primary audiences: researchers and educators. Our toolkit facilitates building data physicalizations and tangible user interfaces with interactive toy bricks.

The toolkit can sense when bricks are added or removed by monitoring changes in capacity, identifying their respective positions in 3D space. Furthermore, the system can detect when bricks are touched. The logs from the performed interactions can be stored, allowing to bridge the gap between digital information and physical interaction with the tangible construction, offering users a more intuitive and immersive experience with the data.

The act of touching a brick serves as a triggering mechanism: if the brick's color has been previously scanned and a touch is subsequently detected, the interaction is interpreted as the addition of a brick. Conversely, if a button has been activated prior to the touch detection, the interaction is read as the removal of a brick (see Figure 2)

In instances where the color of the brick has not been scanned, nor has a button been engaged prior to touch detection, it can serve as a trigger for a specific action, particularly in the context of Tangible User Interface (TUI) prototyping. For instance, touching a tower on one of the plates can activate powering the LED. Our system consists of dedicated 3D-printed parts: bricks and a base plate. To allow for seamless integration with electronics, the printouts were manufactured using two filaments: standard, colorful PLA and conductive composite.

The base plate is equipped with capacitive touch sensors, which are connected to conductive plates arranged in a grid.

Additionally, the system includes a color sensor which allows to recognize the color of the brick. This color scanner is placed in a dedicated slot on the baseplate, separate from the main grid that detects the position of bricks (see component d) in Figure 2). Information about the color might be used for data physicalizations. For example, different colors might indicate different units of measurement.

Our toolkit is based on the Arduino UNO microcontroller, allowing users to customize functions through code. The main features of the ProtoBricks toolkit include logging the building process, creating connections between the structure and visualizations, or interfacing with Arduino-controlled outputs. Finally, the set comprising ProtoBricks can be extended with additional optional software extensions and additional hardware parts. Those include a brick distributor, and software features based on machine learning or allowing for wireless communication with the system.

3.1 Design requirements

After reviewing existing solutions and their limitations (see section 2.1), we determined design requirements for the toolkit, with the goal that is facilitates data physicalization and prototyping of TUIs. First, we wanted to implement (1) relatively simple hardware and software solutions without the need to employ external apparatus (vision systems, tabletops, etc.) that would allow for easy and quick replication with basic knowledge of electronics and 3D printing. Ensuring the system is hidden inside the base allows for a more efficient implementation reminiscent of building on a classic base of bricks. Another aspect was to ensure (2) easy reconfiguration and expansion of the system to adjust it for different, specific needs. Moreover, (3) we wanted to make our toolkit compatible with the popular LEGO® bricks, available in many shapes and colors that could enrich the constructions made with our toolkit. On the software level, we wanted to provide (4) flexible, transparent solutions that can be used in the basic version, as well as easily expanded with custom algorithms, thereby accommodating users with different skill sets. Finally, we wanted to build our toolkit from available and

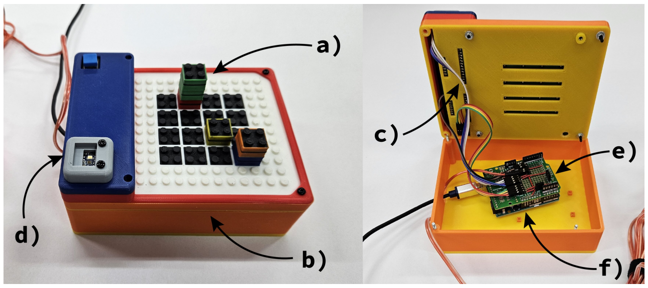

Figure 3: Overview of all parts of the ProtoBricks toolkit.

affordable materials, ensuring the system's compatibility with a wide variety of platforms.

The ProtoBricks toolkit includes the following parts shown in the diagram in Figure 3:

(a) set of dedicated **3D-printed bricks** made from two types of materials (classic and conductive);
(b) **3D-printed baseplate** with its case made out of two types of materials (classic and conductive);
(c) **custom PCB** board placed beneath the baseplate;
(d) **color scanner** with a prepared slot designed for brick placement;
(e) **shield for the Arduino UNO board**;
(f) **dedicated software**.

In order to facilitate understanding and replication, we provide a series of resources along with ProtoBricks. These consist of a repository[1] containing essential files for recreating the system, along with extension elements. The materials include editable schematics and PCB designs for the Eagle[2] software, executable files for printed circuits directly intended for ordering (Gerber), a list of electronic components, and STL files of individual models. We also offer files containing the instruction of assembly of all base printouts and parts, including PCBs and mounting elements, in the Fusion360[3] format (.f3z), as well as fully functional code for Arduino UNO Rev3[4] that enables the recognition of six basic colors. Additionally, we provide three trained neural networks (action recognition, automatic brick scanning, color recognition) designed for use with the Arduino Nano 33 BLE[5] board, STL files for brick's dispenser along with assembly in .f3z format, code for controlling automatic brick feeding, and an archive containing models in .stl format and PCB designs from the first two iterations.

All iterations of developing the ProtoBricks toolkit are presented in the Appendix A.

3.2 Quick start—assembling the toolkit

In the following section, we describe the requirements and steps to be followed to quickly build a system based on open-source resources available on our GitHub. The workflow allows creating a prototype without the need for advanced technical education in the field of PCB design or 3D modeling. However, in order to build the toolkit, basic knowledge of electronics, 3D printing, and programming is required to understand and use the available code.

The process consists of seven steps in four sections (see Figure 4)—preparing the hardware first, then uploading the code and calibrating the system for further programming and use of the toolkit.

Connecting the individual parts is done with screws to ensure easy access to the microcontroller and electronics. The exception is the base plate with 2x2 plates that require gluing. During assembly, glue is applied to the bottom tabs in the base. Then, the plate is held to the edge points of the base plate or previously mounted elements with a top brick to avoid spacing changes resulting from the clearance included in the 3D printing model's design.

3.2.1 Hardware. The prepared models and designs have been adapted to the Raise3D E2[6] printer and to the PLA materials offered by Raise3D[7]. Moreover, Proto-Pasta conductive material[8] and Polymaker PolyLite PLA green material[9] was used. The customized code we provide for each of these colors makes it a recommended solution for makers without specialist experience.

The 3D printing process includes a modular cassette consisting of 5 printouts made of standard material like PLA and 16 plates of conductive filament: (1) the top cover, which covers the touch sensors and contains a place for mounting the button and color sensor, (2) the base plate, enabling the installation of conductive pads in accordance with the designed dot spacing, (3) the color sensor cover with a place for repeated placement of bricks for scanning, (4) 16 2x2 conductive plates that are responsible for transmitting the capacitance values of the bricks placed on them, (5) a shelf holding the PCB board together with the base plate (number 2 above) mounted on it, (6) the base chamber for the microcontroller with an only USB cable and wires connected to button to start the process of removing bricks. In addition, it is necessary to print bricks – as many as needed – in a two-material printing process from a standard and conductive filament (see Figure 6).

The next step is soldering the PCB with electronics and connecting it to the 3D printouts. A characteristic feature of this tool is its flexibility and ease of adaptation to various projects. Therefore, the primary way to connect parts is to use connecting cables with BLS plugs, gold pins, and a shield for Arduino or breadboard. On the surface of the dedicated PCB, there are sockets for two 12 buttons, MPR121 capacitive touch sensors, an I2C bus for connecting to the Arduino microcontroller, and 16 pass-through pads dedicated to connecting 3D printing with the electronics layer. We calculated the number of required sensors by dividing the number of plates by 12 (the number of pads on each sensor), always rounding up. Due to the use of two identical sensors on one I2C bus, it is necessary change the address of one of them, which has already been implemented by default on the board. Sensors have two separate addresses—one default 0x5A and one changed to 0x5C. Goldpins are soldered in a standard way for the THT method, but in the case of connecting the PCB with conductive material, it is necessary to prepare 16

[1]https://github.com/JuliaDominiak/ProtoBricks
[2]https://www.autodesk.com/products/eagle/overview
[3]https://www.autodesk.com/products/fusion-360/overview
[4]https://docs.arduino.cc/hardware/uno-rev3
[5]https://docs.arduino.cc/hardware/nano-33-ble
[6]https://www.raise3d.com/products/e2/
[7]https://www.raise3d.com/filaments/pla/
[8]https://proto-pasta.com/pages/conductive-pla
[9]https://polymaker.com/product/polylite-pla/

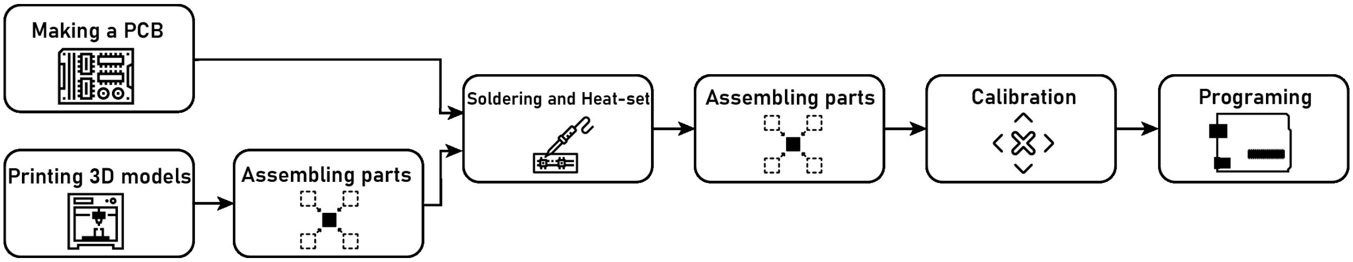

Figure 4: Workflow for constructing the ProtoBricks toolkit based on provided materials.

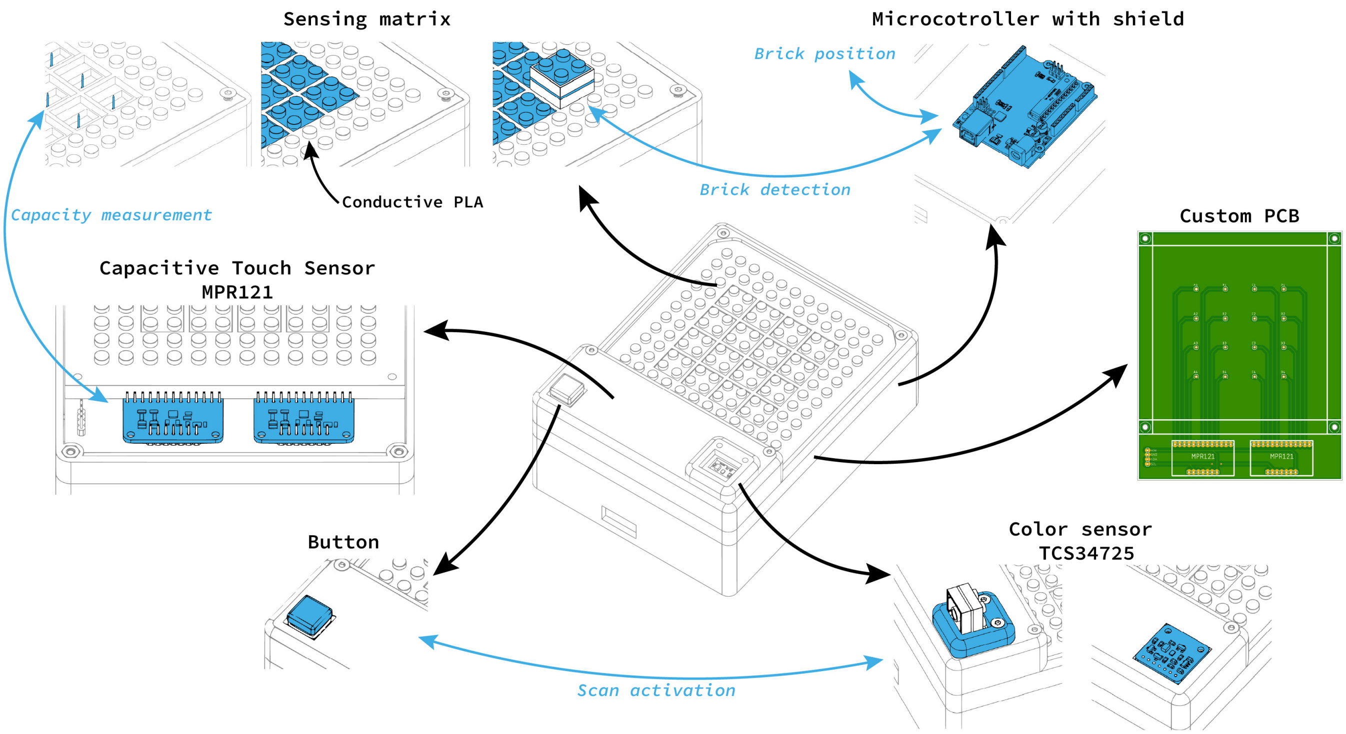

Figure 5: A diagram showing the connections between individual printouts and parts of the toolkit.

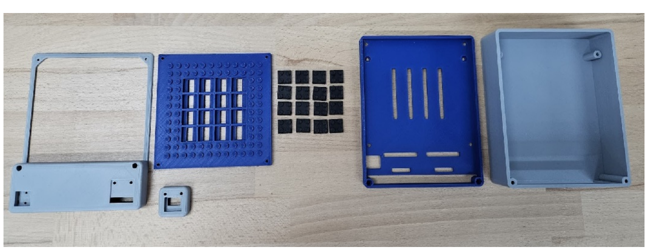

Figure 6: The parts necessary to construct the base made using 3D printing technology.

approximately one-centimeter lengths of silver-coated copper wire. After attaching the PCB and the 3D base with conductive pads using screws, we place a piece of wire in each of the 16 vias, and then, after heating it with a soldering iron, we melt the wire into the conductive plates, using the heat-set insert[10] technique. After connecting the wire, we solder it to the PCB, being careful to not to overheat the structure, nor cause the wire to move in the plastic, and cut off the excess. The connection of parts with the Arduino can be achieved using a breadboard or a universal shield dedicated to the Arduino. In both cases, it is important to connect to the I2C bus—from the main board, the touch sensors, and directly from the color sensor, two buttons—one from the color sensor and the other from information about removing bricks. In the shield, there is an option to lead out all unused pins, which allows the connection of additional electronics, such as LEDs, screens, or sensors. The last stage is to connect all the parts and close the entire system in the housing.

[10]https://ultimaker.com/learn/how-to-use-heat-set-inserts-to-securely-fasten-3d-printed-parts/

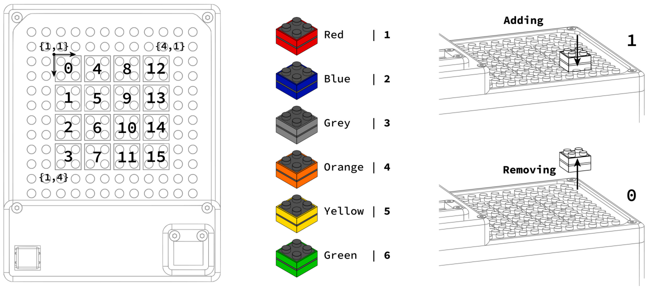

Figure 7: A cheat sheet containing the visual legend of the position in the array and matrix, color numbers, and the value of the digit for adding or removing a brick.

3.2.2 Software. The prepared code is compatible with both the first and the second version of the Arduino IDE and consists of seven files: the main program that initializes all the necessary files and settings (Brick_plate_mini.ino); the program responsible for checking the I2C connection with touch and color sensors (MPR_connections.ino); programs for displaying raw readings from touch sensors in a serial port monitor, used to calibrate thresholds and check connections between the conductive filament and PCB (cap_1.ino, cap_2.ino); a program that acquires RGB values from a color sensor and then compares the values of individual dyes to the defined ranges for each of the six colors and returning a number (color.ino); a program responsible for detecting interactions (both adding and removing bricks) and assigning location numbers (location.ino); additional code for decoding values from location.ino to XY coordinates, returning two values (decoding.ino).

After completing the calibration, one can proceed to use the code. Upon disabling the display of textual information and calibration verification values, the program outputs four comma-separated values in the Arduino's serial port monitor. The first two represent the XY coordinates, followed by a number corresponding to the implemented colors and a variable indicating whether a brick was added or removed. Figure 7 presents a "cheat sheet" containing the direction of position value incrementation, numbers corresponding to individual colors, and an interaction visualization corresponding to the last value. We decided not to implement a GUI, as it would have to be adjusted to the particular context of use. ProtoBricks allows for easy extensions of the toolkit's software, including implementing a dedicated GUI for specific needs.

Furthermore, within the code, it is possible to return a single location value within the range of 0–15 (the locations of individual fields are also depicted in Figure 8) using the location() function. Additionally, it can display the color name immediately after performing a scan to verify the measurement accuracy by uncommenting line 51 in the color.ino file.

3.3 Extension and modifying

The ProtoBricks toolkit was designed to facilitate expansion and modification on both hardware and software levels. However, we emphasize that modifying the toolkit goes beyond the presented Quick Start 3.2 and requires specialist knowledge.

3.3.1 Hardware. The fundamental expansion capability of the system lies in altering the dimensions of the base plate. To accommodate the dimensions of a matrix composed of 2x2 plates requires a change in both the 3D model and the PCB circuit design. Both files are available in an editable version within the GitHub repository[11]. As part of our experimental scope, we proceeded to increase the matrix from 4x4 to 8x8. The most significant change introduced in the model was relocating the microcontroller to the same level as the PCB board, allowing for material usage minimization and increasing the number of touch sensors, which necessitated the use of an I2C multiplexer.

Another crucial aspect to consider is the number of capacitance sensors employed. When utilizing solutions available in the market, we have options due to the quantity of channels—1, 12, or 30. In this scenario, depending on the matrix's size and planned higher number of sensors, it needs to solve duplicated addresses by (1) modifying the hardware address, (2) modifying the software address, or (3) implementing an external multiplexer to expand the quantity of independent I2C buses.

Furthermore, while we employed Arduino Uno as the controller for ProtoBricks, different consumer-grade and custom microcontrollers can be embedded into our toolkit. Therefore, it is possible to make our system wireless (by employing a microcontroller with Wi-Fi or Bluetooth communication), increase the processing power (by choosing a microcontroller with different computing parameters), or fulfill more specific requirements. An example of such expansion by connecting the toolkit to a Raspberry Pi is presented in an article showcasing an intergenerational, remote brick-playing system [56] (see Section 4.1). Finally, ProtoBricks toolkit offers extensive possibilities for integrating sensors and feedback parts (such as vibrational motors, screens, LEDs, and speakers) based on specific needs.

3.3.2 Software. While the whole software is fully editable, we highlight a few directions to extend and modify it, which can prove especially useful, considering the purpose of ProtoBricks:

- Adapting the program to the chosen number of colors of the elements and the dimensions of the base plate's matrix.
- Extending the program to support additional electronics, sensors, and communication.
- Changing the way values are printed or enriching the graphical user interface layer (using for example Processing, Python with PySerial and Tkinter/PyQt, or Node.js with Electron).
- Connecting the GUI to a display device allowing for visual data presentation in various environments (integrate with display devices such as screens, projectors, or LED matrices).
- Ensuring wireless communication through communication modules for Arduino or by establishing connections with Raspberry Pi.
- Modifying the brick detection pipeline, increasing filtration, or enhancing sensitivity to signal changes.

3.3.3 Dispenser. We also developed an optional extension to the ProtoBricks toolkit—a dispenser that allows for serving individually 2x2 brick-sized printouts in a chosen quantity. This extension was

[11] https://github.com/JuliaDominiak/ProtoBricks

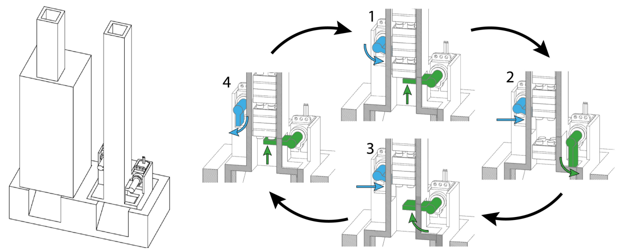

Figure 8: Construction of the dispenser with a housing and a servo mechanism with dedicated horns.

distributed to researchers who used it in a study on physicalization of personal data to enhance reflection, described in Section 4.2.

The dispenser's structure is entirely composed of 3D-printed parts. Inside the chamber, up to 20 bricks can be held. The dispensing process is carried out by two servo mechanisms equipped with dedicated horns, forming the dispensing system. When the lower releasing gripper opens to dispense a single brick, the upper gripper holds the column of elements in the chamber, preventing them from spilling (see 1 and 2 on Figure 8). After delivering a single brick, the trapdoor system closes (see 3 on Figure 8), and the upper servo mechanism releases the column of bricks (see 4 on Figure 8), allowing for the start of the next cycle.

The software initiates the dispensing process after entering a single number in the serial monitor or using multiple dispensers; the user can input a sequence of numbers separated by commas. The algorithm is also equipped with a calibration mode to ensure the system's reliability.

In our GitHub repository[12], we provide both 3D models compatible with micro-sized servos like the PowerHD HD-1810MG[13], as well as software prepared for the Arduino Uno Rev3 microcontroller[14], which coordinates the operation of the servo mechanisms through an 18-channel Mini Maestro servo controller[15].

3.3.4 Artificial neural networks. During the evaluation phase (see section 5.4.1), our participants suggested implementing automatic detection of adding and removing the bricks and automatic color recognition (without the need to scan the bricks or press a dedicated button). Some participants ideated on implementing machine learning algorithms to automate these interactions. To explore this possibility, we trained three neural networks based on our prototype to manage specific tasks related to brick manipulation and recognition. The first network was designed to detect added and removed bricks by analyzing capacity readings over two-second intervals, categorizing the data into labels: addition, removal, or no action. This allowed for precise tracking of brick configuration changes. The second network focused on detecting the placement of bricks on a color scanner, utilizing readings from a color sensor without additional lighting to preserve ambient light conditions, with labels for the data including placement of brick, shadow, or no action. The third network, responsible for recognizing brick colors, used controlled lighting to enhance the detection of pigments from the sensor readings, categorizing the data into specific color labels such as red, orange, yellow, blue, green, and gray. This strategic approach enabled a robust system capable of handling different aspects of brick manipulation and identification efficiently. As this was just an attempt to train the networks for the first two cases, we collected 100 samples for each label, and for the last case, 50 samples. Considering the requirement for simplicity in modifying and expanding the toolkit, we used the online application Edge Impulse[16] to develop the neural network, which offers cloud-based work with a user-friendly interface. After gathering the necessary datasets, we uploaded them to the project and divided them into typical proportions—60% training data, 20% validation data, and 20% testing data. Then, the program analyzed the data in terms of attributes (characteristic features for each label), and vectors were generated for each data point based on these attributes as input signals for the neural network. Due to the defined labels, we applied a classification network, returning the probability of belonging to specific classes. The accuracy of the obtained classification for individual neural networks was 97.9% in the case of distinguishing the adding or removing of a brick (including the neutral state), 95.2% in the case of distinguishing the placement of a brick when scanning the color and shadow resulting from the operation of arms over the base and the surroundings, and 100% when distinguishing six colors prepared in the toolkit. The developed neural networks and the library were also published in a shared repository.

4 DEMONSTRATIVE EVALUATION

We investigated the capabilities of ProtoBricks as a tool for user studies on data physicalization and rapid prototyping of TUIs by adapting *Demonstration* as toolkit evaluation strategy described by Ledo et al. [37]. This method was proved efficient in other toolkit contributions in HCI [36, 42]. To demonstrate the capabilities of ProtoBricks, we distributed our toolkit to two research groups who used it in their respective research studies. The studies explored two different applications: remote inter-generational interaction between children and grandparents and supporting self-reflection by physicalizing the data about one's health. In the latter, our toolkit was extended with the bricks dispenser. Here, we describe the use of the ProtoBricks toolkit in these endeavors to exemplify how the toolkit can be utilized in future HCI research studies.

4.1 Case 1: ProtoBricks as a research probe to study intergenerational remote play

Stefanidi et al. [56] investigated how interactive toys can support intergenerational distributed play. They conducted a within-subject study with 6 pairs of grandparents and their grandchildren, comparing two conditions: playing over distance with regular toy bricks (RB) and with MagiBricks (MB), which utilises our ProtoBricks toolkit. Participants took part in both structured (create specific structures) and unstructured play (interact with the toy bricks freely). During the study, the participant pairs interacted with the toy bricks while being in different spaces and could communicate through video calls. In this study, our toolkit was extended with a feedback system, including a server-client model for communication between the ProtoBricks toolkit that each participant interacted

[12]https://github.com/JuliaDominiak/ProtoBricks
[13]https://www.pololu.com/product/1047
[14]https://docs.arduino.cc/hardware/uno-rev3
[15]https://www.pololu.com/product/1354
[16]https://edgeimpulse.com/

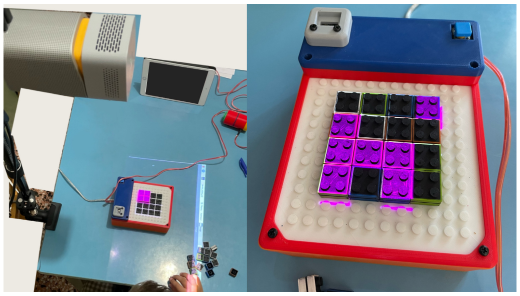

Figure 9: The ProtoBricks toolkit as used in the study with pairs of grandparents and grandchildren in the context of remote intergenerational play.

with, as well as a portable mini-projector that provided visual and audio feedback. In particular, in the MB condition, ProtoBricks detected the placement of bricks and allowed transferring that information between the players through the aforementioned server-client system. Player 1 could experience feedback from Player 2 (and vice versa) in the form of sound and color. On top of a baseplate, the projector displayed green in the corresponding place when a brick is added, red when a brick is removed, and pink for showing all the current positions where player 2 has placed a brick (see Figure 9). Thus, Player 2 had an overview of any changes that Player 1 made on their baseplate for the entire duration of play. Our toolkit allowed for real-time detection of changes regarding what participants constructed upon the baseplates. Tracking the current position of the bricks allowed for displaying dynamic feedback, creating an engaging and tangible experience of remote intergenerational play. The study found that playing with interactive toy bricks enhanced the communication and collaboration between participants, leading to increased feelings of connectedness while playing over a distance. Moreover, the findings suggest that tangible artifacts enhanced with feedback can support children's empowerment and positively affect collaborative activities.

The process and findings of this study demonstrate how the ProtoBricks toolkit could be used as a research probe to investigate tangible interfaces. The form of toy bricks provided affordances familiar to both children and grandparents, making the interaction natural and easy. Moreover, the capability of tracking the position of the individual bricks allowed for displaying dynamic visual and audio feedback. While in this study interaction data was not collected or stored, this capability of the system could further support the analysis, providing information about each interaction with a timestamp.

4.2 Case 2: ProtoBricks as a tool for data-based reflection

Using our toolkit, Bentvelzen et al. [9] investigated whether building tangible representations of health data can offer engaging and reflective experiences. They conducted a between-subject study with N = 60 participants who physicalized their immediate blood pressure data in relation to medical norms. The study compared three conditions: using a standard mobile app, building data representations from toy bricks with instructions, and completing a free-form brick build. In the *instructional* condition, participants used a bricks dispenser—an extension to our ProtoBricks toolkit whose implementation details are described in Section 3.3.3. One dispenser was employed for each of the four bricks' colors. It provided participants with the exact amount and color of bricks that they needed at the moment, according to the instructions. The main motivation behind using the dispenser was to limit the presence of the experimenter—they would have to count and provide the bricks, making the condition appear more social than the use of the mobile app or a free-build. The study revealed that the degree of control is a determining factor for designing physical data representations in personal informatics. The study showed that while free-form conditions necessitated extra time to complete and lacked usability, instructional conditions supported with the brick dispenser fostered focused attention and comparison.

In this study, the bricks dispenser, being an extension to ProtoBricks toolkit, was used to serve researchers in conducting a user study on self-reflection with data physicalization. The presence of the dispenser reduced the bias of instructional conditions being perceived as more social and provided the users with an easy-to-use device, aiding the process of data physicalization.

5 USAGE EVALUATION

As described by Ledo et al. [37], usage evaluation allows studying how users appropriate the toolkit, investigating whether the toolkit is conceptually clear, easy to use, and valuable to the audience. This evaluation method was proven successful in previous toolkits-based research [7, 38], often in combination with the *Demonstration* technique [54, 68]. We therefore opted for a similar approach. Inspired by previous endeavours [31], we conducted a two-level *Usage Evaluation* consisting of (1) workshops with students who used ProtoBricks during their classes and (2) expert interviews with educators and researchers. This approach corresponds to the key aspect of scaleness for ProtoBricks.

5.1 Workshops

To elicit early feedback and user experiences of using ProtoBricks for data physicalization and TUI prototyping and investigate which usage scenario holds a greater potential in supporting creativity, we conducted a series of participatory workshops with students. We used participatory techniques as they allow for juxtaposition of perspectives, ensuring triangulation [2, 46], contributing to a more credible and trustworthy inquiry [22]. Moreover, this approach was used in previous toolkit studies [31]. During workshops, students of varying experience with electronics and programming completed two tasks using ProtoBricks: data physicalization and TUI rapid prototyping. They were observed during these tasks and prompted about their experience. Each workshop took about 2h. Moreover, at the end of each task, they filled CSI questionnaire [12]. This tool was designed to evaluate how well a tool supports creativity which corresponds which one of our requirements for the toolkit.

5.1.1 Participants. We conducted three sessions of workshops with a total of N = 11 participants (5 males, 6 females, aged 19–26, M = 22.90, SD = 2.34). There were 3–4 participants present at each workshop. Participants of these studies were bachelor's and master's students affiliated with the Human-Computer Interaction field

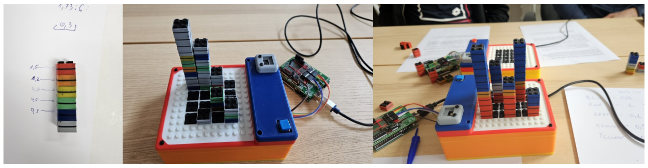

Figure 10: An example legend of the value distribution along with a physical visualization of data regarding power consumption during the first task of the workshops.

from a university located in central Europe. As the study was conducted during extra-curricular classes, there was no remuneration provided, in line with local regulations. Table 1 shows the overview of participant's profiles.

5.1.2 Workshop Structure. At the beginning of the workshop, we obtained informed consent and distributed a short survey on participants' subjective experience with electronics, microcontroller programming, and interest areas in prototyping at large.

Then, one researcher introduced the ProtoBricks toolkit and explained the provided software and hardware and the toolkit's capabilities. Afterwards, participants were provided with instructions for Task 1: *Data Physicalization.* Students were instructed to physicalize data of one month's electrical energy consumption in a studio apartment. The data attached to the task was an anonymized record of the electricity usage of one of the authors. The data was provided in the form of an Excel spreadsheet containing monthly records of electricity consumption in kilowatt-hours (kWh) for each day, along with information about the hour and value of the highest peak consumption. For this task, participants were provided with dedicated conductive bricks in 6 available colours (red, green, yellow, grey, blue, orange). Moreover, they were free to use Excel for calculations or perform the necessary calculations by hand when needed. Participants had full freedom in deciding which data and characteristics they wanted to present and how they would visualize them. When participants were ready to show their visualization, students completed CSI surveys [12]. Then, the researcher took photos of the construction, turned on the audio recording, and conducted a short group interview in the form of a focus group to delve into their perceptions regarding the physical representation of data – if and how they found this approach suitable, whether they identified advantages of such representation, and if they envisioned alternative data for presenting in this form and the benefits associated with them. Moreover, students were prompted about the system—whether it is easy to use, whether they would use it in different contexts, and what improvements they would suggest.

Afterwards, participants were provided with instructions for Task 2: *Rapid prototyping* of the tangible smart home interface. Due to space limitations of the base plate, we suggested designing a studio apartment with basic furnishings. This time, participants had access not only to dedicated bricks but also to original LEGO® bricks of various shapes and colours, enabling a more straightforward visual presentation of rooms. 3D-printed conductive bricks were used as control elements, such as switches, volume, or position control in the room. Meanwhile, output visualizations were created using Arduino-compatible electronics, such as LEDs, servo motors,

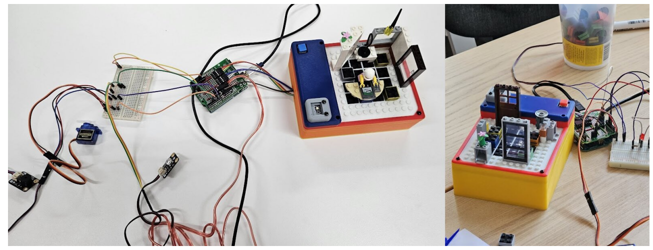

Figure 11: Sample mock-ups of a studio apartment with the implemented hardware side of the electronics connected to Arduino during the second task using the system as an IoT interface.

buzzers, screens, or vibration motors. This task facilitated verification of the integration between the input layer represented by the bricks and the output layer in the form of electronics, using standard code functions. After completing the task, participants were again provided with CSI surveys. Again, the researcher conducted a short group interview in the form of a focus group concerning the usage of the system and the presented control panel. The interview was audio-recorded, and the constructions made by the students were photographed.

5.2 Expert interviews

To complement the evaluation of ProtoBricks, we conducted a series of expert interviews to define potential areas of system application and target users in education. Moreover, this stage allowed us to contrast the experiences of younger, less experienced users with views of older participants with more years of experience in microcontroller-based systems.

5.2.1 Participants. We recruited five experts who had at least five years of experience in conducting classes with students (4 male, 1 female, aged 27–47, M = 37.60, SD = 7.09). No remuneration was provided for their participation in the interviews. Table 2 provides a detailed description of the participants' profiles, their experience with microcontroller-based systems, and the types of projects they were involved in. The participants had varying levels of experience in prototyping. Except for one person, all participants designed and programmed their own systems.

5.2.2 Procedure. First, the participant signed an informed consent form and filled in a short survey concerning their academic experience and technological proficiency. Then, we presented the ProtoBricks toolkit, describing its functions, capabilities, hardware, and software solutions. Participants could play with the toolkit, test the provided code, and see logs available in the program. After familiarizing themselves with the toolkit, a researcher asked questions about their overall impressions regarding the aesthetic, technical, and usability aspects of the system. Additionally, they were inquired about their opinions on the potential difficulties of using the toolkit in its current form and at what stage of education the system could be integrated and in what form. Next, we moved on to questions about the possibilities for changes and adaptations they envisioned for younger users with less programming experience. Furthermore, we prompted them whether and how they would use

Table 1: Participant profiles in the interviews, including their current level of study prototyping experience (5-item Likert scale) as well as the purpose of creating prototypes and types of projects

ID	Age	Gender	Current deg.	Prot. exp.	Purpose of prot.	Types of projects
P1_1	25	male	Ph.D.	2	professional	subject, thesis
P1_2	24	female	MSc.	1	education	subject, research projects
P1_3	22	male	Eng.	0	-	-
P1_4	19	male	Eng.	1	hobby	-
P2_1	23	male	MSc.	4	hobby	automation, entertainment/decoration, thesis
P2_2	20	female	Eng.	1	education	lifestyle, competition
P2_3	20	male	Eng.	2	hobby	entertainment/decoration
P3_1	26	female	MSc.	1	education	subject
P3_2	25	female	MSc.	2	education	entertainment/decoration, subject
P3_3	24	female	MSc.	2	education	automation, subject
P3_4	24	female	MSc.	2	education	subject

Table 2: Participant profiles in the interviews, including their prototyping experience (5-item Likert scale) as well as the purpose of creating prototypes and types of projects

ID	Age	Gender	Y. of exp.	Prot. exp.	Purpose of prot.	Types of projects
E0	38	male	11	2	professional	automation, entertainment/decoration, subject
E1	27	male	5	3	professional, hobby	automation, entertainment/decoration, subject
E2	38	female	12	2	professional	subject
E3	38	male	14	4	professional	entertainment/decoration, research projects
E4	47	male	20	2	professional, hobby	automation, lifestyle, entertainment/decoration

ProtoBricks to physicalize data or prototype IoT interfaces in their everyday life. Finally, they were asked whether they saw any limitations and what their general impressions were regarding tangible systems.

5.3 Analysis

All collected data encompassing the group interviews conducted during the workshops and the expert interviews were recorded and then transcribed verbatim (total duration 2 hours and 37 minutes). We applied a pragmatic approach to thematic analysis [11]. Due to the limited number of interviews, we established a preliminary coding framework through open coding of all interviews, which were divided among three researchers. In a joint discussion session, a preliminary coding tree was constructed. All interviews were subsequently split between two coders and analysed a second time using the preliminary coding tree. During a final discussion session, we refined the coding tree and identified recurring themes in the data.

5.4 Results

In this section, we present the results of our study, divided into three main themes developed during the thematic analysis process: *Creativity*, *Limitations*, and *Interaction*. We have supported and illustrated the results with quotes, highlighted in italics, and labelled with participant or expert numbers. Within all the themes, we present the study participants' accounts, which showcase common and contrasting attitudes, aiming to provide comprehensive insight into using the ProtoBricks toolkit. Moreover, all works constructed during workshops with students are presented in the Appendix B.

5.4.1 ***Creativity***. During interviews with both workshop participants and experts, a variety of diverse potential applications emerged, encompassing both education and entertainment, as well as data physicalization. The participants themselves noticed the multitude of ideas suitable for a wide range of target groups.

Physicalization The application of various forms of physical representation of numerical data was highlighted by both groups participating in the study. The starting point for the discussion was the proposed task, which also had a connection to the physicalization of data related to pulse and heart rate [9]. The most common types of data to visualize that emerged during our studies were financial tracking (house budget, types of expenses, savings, currency exchange rates). One of the workshop participants commented that they would use the toolkit to track their personal spending.

> *(P1_1) Tracking your household budget would also be easy with this, as the colors would indicate what you're spending on.*

Moreover, some participants suggested using SmartBricks as a means to manifest socially important topics, encouraging reflection. Our participants ideated on installing visualizations of the amount of waste thrown every day, food waste, and meat consumption in schools and other public spaces. As an alternative to digital counterparts and inputting data into electronic systems, especially in preschools and primary schools, points for activity and task division in calendars were considered. Drawing inspiration from the second task related to I/O programming, the toolkit found its application in controlling smart homes (including a control room for children), physical space design for individuals unfamiliar with 3D room design, and in the interface for managing access to rooms. **Algorithmics/Programing** The construction of code and the structure

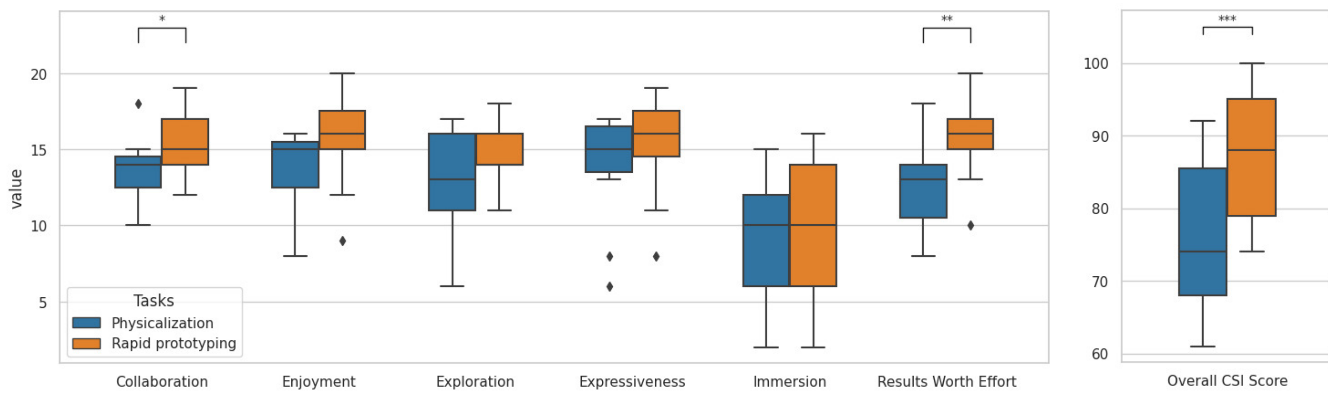

Figure 12: The distribution of CSI scores and its subscales across conditions: Physicalization and IoT. Significant pairs are marked with annotations. A single asterisk denotes a significance of (p = .055*2), a double asterisk denotes a significance of (p = .010*2), and a triple asterisk denotes a significance of (p = .017*2)

of the toolkit have been extensively discussed in various application scenarios, especially by experts who see its application at all stages of coding and electronics learning. Starting from preparing brick coding and wiring connections to advanced stages of microcontroller programming and user interface development, including graphical user interfaces (GUIs) for different application scenarios. One expert also ideated on using ProtoBricks as the tool to teach the basics of UX design.

> *(E2) Well, that's why the best solution would be to make it a good tool for learning, for example, as a task to design and implement an interface. And it could also lead to some additional ... introduction to, for example, UX at this point, doesn't it?*

Two experts also suggested using the toolkit to teach computational thinking, for example, by using smart bricks as interactive puzzles, such as the Tower of Hanoi. However, experts emphasize that to fully use the potential of this toolkit, advanced knowledge above the basic level associated with the Arduino platform is required. One expert suggested that tangible bricks could be used to represent the values in an artificial neural network (e.g., weights, number of neurons, etc.). They elaborated that such tangible representations could help students to understand machine learning. Some experts also noted that in the early stages of education for younger children, the toolkit can serve as a tool to make the classes more engaging by physically representing values or by using bricks for voting in quizzes. One expert also noted that ProtoBricks could help to visualize syllables in the early stages of grammar learning.

Entertainment In addition to offering the toolkit for remote play [56] and integrating it with playful education, experts noticed the broad potential of the toolkit as a universal platform for playing board games, both in hybrid form and for remote gaming between players in distant locations. Thanks to the physical form of the bricks, they do not disrupt immersion, unlike screens or digital board game versions, creating a "bridge" between online gaming and the traditional form of board games. In this form, participants envisioned games such as chess, tic-tac-toe, battleships, or Mastermind. In the case of local games, the baseboard with its bricks could serve as statistic markers in RPGs or as interactive tokens during poker games with automatic counting values.

> *(E0)But on the one hand, you have physical cards, where there is a mess, and on the other hand, you have it in a purely digital version, where it disturbs this immersion, and if we use these bricks, then we are perfectly in between a board game and a digital one.*

5.4.2 *Limitations.*

During both workshops and expert interviews, many aspects related to technical expectations of the system emerged.

The key issue raised by all individuals from both groups was the insufficient base plate size. During workshops, particular attention was paid to this issue, especially concerning tasks related to physicalization. The opinions were related to the constraints of this aspect, often requiring a choice between trends and numerical values, which prioritized surface requirements over the achievement of the intended effect or the relevance of the parameter and trend. This led to the neglect of data or a less logical arrangement of data from the participant's perspective. The compact dimensions also hindered the readability of bricks in the middle of the workspace and made it difficult to operate within the structure, especially during creating and presenting tasks related to I/O. During the workshop, general opinion surfaced about the need to increase the size of the base. Experts were inclined not only to increase the dimensions but also to change the shape and placement of the sensor pads.

> *(E4) Maybe my thinking is a bit limited cause it's only four by four.*

Most experts have emphasized the need to implement feedback in the form of a small screen, LED matrix, or RGB LEDs to confirm actions such as placing a brick or scanning a color. They also highlighted that it offers the possibility to develop a more advanced graphical user interface without the need to connect it to a computer. One expert emphasized that in the case of frequent use of the color scanning function, the immediate display of information within the prototype would work better.

(E4) "Well, well, I often need this [prototype], don't I? Because later, well, when I place this block [brick], even if it's taught [programmed], and it doesn't work, I don't know what's happening, right?"

Concerning technical aspects associated with the production of bricks, there was a recurring opinion regarding color saturation and soiling caused by the thin color layer compared to the black, conductive core. Participants noticed an issue with the thin layer of colored material around the black core, which resulted in a significant reduction in the aesthetics of the brick.

(P1_2) I mean, I don't think there's enough material; it's too thin and too see-through, this black is visible...

During the workshops, participants also noted the lack of perfect fit and the bricks tending to "laying" on top of each other rather than properly connecting. The number of available colors generally satisfied the participants. However, they pointed out that for more complex tasks, especially those related to physicalization, a greater variety would always be advantageous. The available colors allowed for both mapping specific visualized values onto bars and were based on associations (e.g., blue—water/air, yellow—light, red—OFF, green—ON). Participants in one of the groups also suggested the possibility of enhancing the bricks's surface with a texture to facilitate better differentiation for visually impaired individuals.

While working with the system, users noticed an issue with the prepared color recognition algorithm, which required frequent rescanning. This was inconvenient, especially when colors were frequently changing.

(P3_1) Those colors were irritating when [the system] wouldn't read them.

Due to the necessity of using the buttons both for signaling the removal of bricks and for scanning colors, participants pointed out the need to change the method of doing so. They suggested implementing automatic color recognition. This idea applies to recognizing signals during interaction on the base plate, as well as recognizing the brick's color after placing the element in the scanner or summing up the tower's heights.

(P1_2) However, as for future expansion plans, it would be automatic color detection; for example, when placing [a brick], the sensor should immediately record [this], and I shouldn't have to press the button.

5.4.3 ***Interaction***. During both student workshops and expert interviews, our participants noted many aspects of interacting with bricks related to their familiar form. Most individuals, both in childhood and adulthood, have experienced associations and confidence in using the developed system while playing with bricks, building them, or collecting them. Participants had no trouble completing constructions during tasks or finding diverse applications for different age groups. They appreciated the versatility of the kit and its compatibility with LEGO® sets available on the market, which expanded the possibilities and aesthetics of the created models. For example, in rapid TUI prototyping task, students used LEGO® flowers to decorate the flat, LEGO® similar to modern furniture or even placed LEGO® figures inside the flat. The ability to operate not only in three-dimensional space through the use of spatial bricks but also to add a fourth dimension through color has been emphasized by experts as an expansion of the operational space.

(E4) Well, it'd be a waste not to take advantage of it [color]. The color, in terms of its position, doesn't seem overly exotic to me, but the height or elevation [of the build], feels like another coordinate or dimension.

Other aspects related to the toolkit highlighted in interviews included the challenges and the necessity of creating simplified versions of the program for less advanced users, engaging the youngest individuals in building and education through play with the toolkit, and providing extensive support for collaboration during activities, especially for fostering creativity. However, the current form of the toolkit was assessed as best suited for quick exposition of general data rather than precise data in the context of data physicalization. However, the current universal code form and technical layers have been assessed by experts as requiring knowledge, which means that the toolkit may not find initial use in the early stages of learning microcontroller programming.

(E0) The bricks are simple, and Arduino is simple, but if you combine one with the other, you have to be a master at both.

6 DISCUSSION

Here, we first present an overview of potential applications of the toolkit, as suggested in the studies. Then, we discuss ProtoBricks' potential in supporting creativity. Following this, we address the toolkit's limitations and consider directions for future development.

6.1 Potential applications

The core motivation for developing ProtoBricks were supporting data physicalization and TUIs prototyping in research and education. Here we reflect on the applicability of the toolkit in relation to these categories.

6.1.1 Physicalization. The ProtoBricks toolkit offers practical means for researchers, educators and designers to explore data physicalization. By working with ProtoBricks, designers can be supported in deciding which elements of their physicalization ought to be interactive, as every brick can be interactive or not.

By analyzing and gathering feedback from both research participants and the researchers who received the toolkit, we can identify the various ways in which ProtoBricks can be applied for designing physical representations. We present some examples below, relating to different fields:

- **Ecology**: As a visual of electricity consumption and waste generation. During the workshops, students created a number of bar chart-like representations of electricity consumption.
- **Spelling and mathematics**: As a visualization tool to teach spelling or simple mathematical operations. In this case, bricks can help to visualise words which are broken down into syllables or simulate numbers. This feature is particularly useful in early school education, aiding students in mastering fundamental linguistic and mathematical concepts through interactive and engaging visual representations, as noted in the expert interviews.

- **Finance**: As a way to track personal finances, monitor household budget and savings, visualize market trends, and provide a holistic view of users' financial landscape. As noted by one of the students, different colors of the bricks could indicate different category of spending (for example, food, clothes, energy, etc.)
- **Well-being**: For monitoring various metrics for analyzing personal health and well-being. For example, it could provide a visual record of meat consumption, weight, blood pressure, and pulse readings.

6.1.2 TUIs prototyping. Through evaluation with research participants and researchers to whom the toolkit was distributed, we can identified key examples of using ProtoBricks in the domain of rapid prototyping TUIs:

- **Smart home and buildings**: A control panel designed for controlling a room. The bricks can be used to build prototypes of augmented everyday objects, which is in line with the needs of researching technology for the home [16]. The toolkit allows for the distribution of access to different rooms through the placement of bricks on a spatial model, functioning as an interactive spatial remote control.
- **Interactive board game or score counter**: A tangible platform for digital board games played remotely, such as chess, tic-tac-toe, battleship, and Mastermind. Moreover, it can function as a score counter for RPG games or poker chips. Experts also noted that the toolkit can serve for younger children as an educational toy: for example they could use bricks to vote for quizzes.
- **Learning programming/electronics**: As ProtoBricks is based on Arduino, it is a promising tool to learn how to program it and connect to other components. The affordances of toy bricks might make it more pleasant form factor for beginner users.

Except the use-cases mentioned explicitly in the studies, we believe that ProtoBricks can be used in a number of other areas due to its adaptable nature and familiar affordances, yet, other applications require further investigations.

6.2 Supporting creativity

As toy bricks are known for they ability to spark creativity [18, 48], we were interested whether ProtoBricks also holds such potential. Observing numerous ideas for using the toolkit in various contexts, we are optimistic that ProtoBricks, sharing the affordances of toy bricks, can spark inventiveness, as using existing objects in different ways for new purposes supports creativity [30]. Moreover, using CSI questionnaire, we investigated which usage scenario (data physicalization or TUI prototyping) seems more promising in supporting creativity. The differences in CSI scores between the two workshop tasks revealed that rapid prototyping of TUIs outperformed the physicalization task in terms of creativity support. Particularly, significant differences were observed in the categories of *Collaboration* and *Results Worth Effort*. We observed that in the TUI prototyping task participants were especially enthusiastic over combining ProtoBricks with the LEGO® bricks and electronic components. Our findings suggest that the prototyping of Tangible User Interfaces through increased complexity allows for better task allocation and more joint collaboration within the group.

6.3 Limitations and future work

The ProtoBricks toolkit, while versatile in its applications, does have a significant barrier in terms of its accessibility. The necessity for programming aptitude became apparent during its usage. Feedback from experts indicated that utilizing the toolkit requires understanding of the Arduino platform. This sentiment was echoed during the workshops. Participant groups voiced apprehensions regarding the intricacy of the tasks. Only the second group, which had a member with an advanced skill level (4) in programming, was able to navigate through the tasks without seeking external aid. This individual's proficiency in coding enabled the group to sidestep major challenges. In contrast, Groups 1 and 3, devoid of members with skills surpassing level 2, frequently required guidance throughout the programming phase.

The toolkit is crafted in a broad-spectrum style, utilizing C/C++ for Arduino compatibility. This design choice ensures that it can be flexibly tailored to various projects, research, or envisioned applications. Nonetheless, the inherent requirement for programming skills can deter potential users. To bridge this gap and cater to the prospective user without a programming background, our future endeavors should encompass the development of a more intuitive user interface. Such an interface would either eliminate the need for direct coding or incorporate visual programming. This way, the toolkit's benefits could be harnessed by a wider audience, thereby increasing its usability and impact. Moreover, considering its open architecture, future iterations could facilitate the incorporation of any electronics compatible with the chosen microcontroller, amplifying its adaptability.

The form of ProtoBricks is not without its set of limitations. Drawing inspiration from the conventional brick form factor, these tools carry the weight of cultural connotations associated with bricks, which are well-known toys. Such cultural context can influence and sometimes restrict the range of conceptualizations and creative applications users might explore. Furthermore, their form is innately discrete. While this characteristic ensures modularity and ease of use in many scenarios, it also implies a fixed granularity in design, potentially limiting the flexibility and continuity in more intricate projects. Therefore, users must approach these tools with an understanding of these constraints, recognizing both the potential and the limitations the brick form introduces to their work. That is why, if the target design is a form of high fidelity, we recommend using our toolkit in the early stages of the design process.

6.4 Sustainability considerations

While ProtoBricks facilitates tangible interface prototyping, it does introduce some environmental considerations, chiefly due to the utilization of 3D filaments and printed circuit boards (PCBs). The use of carbon filament, although necessary to provide connectivity to the bricks, has ecological implications. We also cannot ignore the energy required to print bricks and PCBs. To enhance the sustainability of ProtoBricks, future iterations could explore incorporating biodegradable or recycled 3D printing materials. Additionally, adopting a modular PCB design, which allows users to replace only the faulty or outdated components rather than discarding the entire board, can minimize electronic waste. We encourage users to repurpose or recycle used bricks, as they are suitable to be used in filament recycling machines. Using our toolkit in the early stages of the design process is also an environmental consideration. Larger deployments of interacting with bricks prototypes should use standard bricks and dedicated electronics to minimize the footprint.

7 CONCLUSION

In this work, we introduced and evaluated ProtoBricks—a versatile and replicable toolkit for tangible prototyping and data physicalization. We provided detailed instructions about the toolkit's fabrication process, tools and materials necessary to replicate the toolkit, and information on how to access and use the open-source code provided with the toolkit. Moreover, we described optional expansions and modifications, including both hardware and software alterations. We also illustrated the possible use of ProtoBricks in future HCI endeavors by presenting two case studies where our toolkit was used in two user studies as a tangible user interface and data physicalization tool. We evaluated the usage of ProtoBricks in workshops with students and a series of expert interviews. Our results indicate that ProtoBricks might facilitate creativity and has the potential to be used in a wide range of scenarios in data physicalization, TUIs, and education. Furthermore, workshops with students and expert interviews helped us to identify space for future improvements and modifications. As we make the toolkit available for the wider research community, we hope that it inspires creative research endeavors.

ACKNOWLEDGMENTS

The project was partially founded by the National Science Center through the Preludium-20 grant program (no. 2021/41/N/ST6/03676).

REFERENCES

[1] Diana Africano, Sara Berg, Kent Lindbergh, Peter Lundholm, Fredrik Nilbrink, and Anna Persson. 2004. Designing Tangible Interfaces for Children's Collaboration. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems* (Vienna, Austria) *(CHI EA '04)*. Association for Computing Machinery, New York, NY, USA, 853–868. https://doi.org/10.1145/985921.985945

[2] Sabreena Ahmed and Ratnawati Mohd Asraf. 2018. The workshop as a qualitative research approach: lessons learnt from a "critical thinking through writing" workshop. *The Turkish Online Journal of Design, Art and Communication* 2018 (2018), 1504–1510.

[3] J. Andersen. 2022. *The LEGO Story: How a Little Toy Sparked the World's Imagination.* HarperCollins. https://books.google.pl/books?id=EslaEAAAQBAJ

[4] David Anderson, James L. Frankel, Joe Marks, Aseem Agarwala, Paul Beardsley, Jessica Hodgins, Darren Leigh, Kathy Ryall, Eddie Sullivan, and Jonathan S. Yedidia. 2000. Tangible Interaction + Graphical Interpretation: A New Approach to 3D Modeling. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*. ACM Press/Addison-Wesley Publishing Co., USA, 393–402. https://doi.org/10.1145/344779.344960

[5] Masahiro Ando, Yuichi Itoh, Toshiki Hosoi, Kazuki Takashima, Kosuke Nakajima, and Yoshifumi Kitamura. 2014. StackBlock: Block-Shaped Interface for Flexible Stacking. In *Adjunct Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 41–42. https://doi.org/10.1145/2658779.2659104

[6] Patrick Baudisch, Torsten Becker, and Frederik Rudeck. 2010. Lumino: Tangible Blocks for Tabletop Computers Based on Glass Fiber Bundles. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '10)*. Association for Computing Machinery, New York, NY, USA, 1165–1174. https://doi.org/10.1145/1753326.1753500

[7] Benjamin B Bederson, Jesse Grosjean, and Jon Meyer. 2004. Toolkit design for interactive structured graphics. *IEEE Transactions on software engineering* 30, 8 (2004), 535–546.

[8] Marit Bentvelzen, Julia Dominiak, Jasmin Niess, Frederique Henraat, and Paweł W. Woźniak. 2023. How Instructional Data Physicalisation Fosters Reflection in Personal Informatics. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 158, 15 pages. https://doi.org/10.1145/3544548.3581198

[9] Marit Bentvelzen, Julia Dominiak, Jasmin Niess, Frederique Henraat, and Paweł W Woźniak. 2023. How Instructional Data Physicalisation Fosters Reflection in Personal Informatics. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–15.

[10] Alexander Berman, Francis Quek, Robert Woodward, Osazuwa Okundaye, and Jeeeun Kim. 2020. "Anyone Can Print": Supporting Collaborations with 3D Printing Services to Empower Broader Participation in Personal Fabrication. In *Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society*. 1–13.

[11] Ann Blandford, Dominic Furniss, and Stephann Makri. 2016. *Qualitative HCI research: Going behind the scenes.* Morgan & Claypool Publishers.

[12] Erin A Carroll, Celine Latulipe, Richard Fung, and Michael Terry. 2009. Creativity factor evaluation: towards a standardized survey metric for creativity support. In *Proceedings of the seventh ACM conference on Creativity and cognition*. 127–136.

[13] Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. 2012. CapStones and ZebraWidgets: sensing stacks of building blocks, dials and sliders on capacitive touch screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2189–2192.

[14] Claudia Daudén Roquet, Nikki Theofanopoulou, Jaimie L Freeman, Jessica Schleider, James J Gross, Katie Davis, Ellen Townsend, and Petr Slovak. 2022. Exploring Situated & Embodied Support for Youth's Mental Health: Design Opportunities for Interactive Tangible Device. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 331, 16 pages. https://doi.org/10.1145/3491102.3502135

[15] Clifford De Raffaele, Serengul Smith, and Orhan Gemikonakli. 2018. An Active Tangible User Interface Framework for Teaching and Learning Artificial Intelligence. In *23rd International Conference on Intelligent User Interfaces* (Tokyo, Japan) *(IUI '18)*. Association for Computing Machinery, New York, NY, USA, 535–546. https://doi.org/10.1145/3172944.3172976

[16] Audrey Desjardins, Ron Wakkary, and William Odom. 2015. Investigating Genres and Perspectives in HCI Research on the Home. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 3073–3082. https://doi.org/10.1145/2702123.2702540

[17] George W. Fitzmaurice, Hiroshi Ishii, and William A. S. Buxton. 1995. Bricks: Laying the Foundations for Graspable User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '95)*. ACM Press/Addison-Wesley Publishing Co., USA, 442–449. https://doi.org/10.1145/223904.223964

[18] David Gauntlett. 2014. The LEGO System as a tool for thinking, creativity, and changing the world. (2014), 215–231. https://doi.org/10.4324/9781315858012-18

[19] Sebastian Günther, Florian Müller, Martin Schmitz, Jan Riemann, Niloofar Dezfuli, Markus Funk, Dominik Schön, and Max Mühlhäuser. 2018. CheckMate: Exploring a tangible augmented reality interface for remote interaction. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–6.

[20] Ankit Gupta, Dieter Fox, Brian Curless, and Michael Cohen. 2012. DuploTrack: a real-time system for authoring and guiding duplo block assembly. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 389–402. https://doi.org/10.1145/2380116.2380167

[21] Mohammad Hadhrawi and Kent Larson. 2016. Illuminating LEGOs with digital information to create urban data observatory and intervention simulator. In *Proceedings of the 2016 ACM conference companion publication on designing interactive systems*. 105–108.

[22] Catherine Houghton, Dympna Casey, David Shaw, and Kathy Murphy. 2013. Rigour in qualitative case-study research. *Nurse researcher* 20, 4 (2013).

[23] Meng-Ju Hsieh, Rong-Hao Liang, Da-Yuan Huang, Jheng-You Ke, and Bing-Yu Chen. 2018. RFIBricks: Interactive Building Blocks Based on RFID. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (, Montreal QC, Canada,) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3173574.3173763

[24] Samuel Huron, Sheelagh Carpendale, Alice Thudt, Anthony Tang, and Michael Mauerer. 2014. Constructive visualization. In *Proceedings of the 2014 conference on Designing interactive systems*. 433–442.

[25] Koshi Ikegawa and Buntarou Shizuki. 2018. Tesla Blocks: Magnetism-Based Tangible 3D Modeling System Using Block-Shaped Objects. In *Proceedings of the 30th Australian Conference on Computer-Human Interaction* (Melbourne, Australia) *(OzCHI '18)*. Association for Computing Machinery, New York, NY, USA, 411–415. https://doi.org/10.1145/3292147.3292221

[26] Koshi Ikegawa, Masaya Tsuruta, Tetsuya Abe, Arika Yoshida, Buntarou Shizuki, and Shin Takahashi. 2016. Lightweight Capacitance-Based Block System for 3D Space Interaction. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces* (Niagara Falls, Ontario, Canada) *(ISS '16)*. Association for Computing Machinery, New York, NY, USA, 307–312. https://doi.org/10.1145/2992154.2996772

[27] Kaori Ikematsu and Itiro Siio. 2018. Ohmic-touch: extending touch interaction by indirect touch through resistive objects. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–8.

[28] Yvonne Jansen. 2014. *Physical and tangible information visualization.* Ph. D. Dissertation. Université Paris Sud-Paris XI.

[29] Jan Joosten. 1999. Denise Schmandt-Besserat, How Writing Came About, University of Texas Press, Austin, 1996. *Revue d'Histoire et de Philosophie religieuses* 79, 2 (1999), 242–243.

[30] H. Kanematsu and D. Barry. 2016. Creativity and Its Importance for Education. (2016), 3–7. https://doi.org/10.1007/978-3-319-19234-5_1

[31] Jakob Karolus, Francisco Kiss, Caroline Eckerth, Nicolas Viot, Felix Bachmann, Albrecht Schmidt, and Pawel W. Wozniak. 2021. EMBody: A Data-Centric Toolkit for EMG-Based Interface Prototyping and Experimentation. *Proc. ACM Hum.-Comput. Interact.* 5, EICS, Article 195 (may 2021), 29 pages. https://doi.org/10.1145/3457142

[32] Kunihiro Kato and Homei Miyashita. 2016. 3D Printed Physical Interfaces that can Extend Touch Devices. In *Adjunct Proceedings of the 29th Annual ACM Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 47–49. https://doi.org/10.1145/2984751.2985700

[33] Scott R. Klemmer, Jack Li, James Lin, and James A. Landay. 2004. Papier-Mache: toolkit support for tangible input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '04)*. Association for Computing Machinery, New York, NY, USA, 399–406. https://doi.org/10.1145/985692.985743

[34] Sébastien Kubicki, Sophie Lepreux, and Christophe Kolski. 2012. RFID-driven situation awareness on TangiSense, a table interacting with tangible objects. *Personal and Ubiquitous Computing* 16 (2012), 1079–1094.

[35] Mannu Lambrichts, Raf Ramakers, Steve Hodges, Sven Coppers, and James Devine. 2021. A Survey and Taxonomy of Electronics Toolkits for Interactive and Ubiquitous Device Prototyping. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 2, Article 70 (jun 2021), 24 pages. https://doi.org/10.1145/3463523

[36] David Ledo, Fraser Anderson, Ryan Schmidt, Lora Oehlberg, Saul Greenberg, and Tovi Grossman. 2017. Pineal: Bringing passive objects to life with embedded mobile devices. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 2583–2593.

[37] David Ledo, Steven Houben, Jo Vermeulen, Nicolai Marquardt, Lora Oehlberg, and Saul Greenberg. 2018. Evaluation Strategies for HCI Toolkit Research. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–17. https://doi.org/10.1145/3173574.3173610

[38] David Ledo, Miguel A. Nacenta, Nicolai Marquardt, Sebastian Boring, and Saul Greenberg. 2012. The HapticTouch Toolkit: Enabling Exploration of Haptic Interactions. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction* (Kingston, Ontario, Canada) *(TEI '12)*. Association for Computing Machinery, New York, NY, USA, 115–122. https://doi.org/10.1145/2148131.2148157

[39] Chi-Jung Lee, Rong-Hao Liang, Ling-Chien Yang, Chi-Huan Chiang, Te-Yen Wu, and Bing-Yu Chen. 2022. NFCStack: Identifiable Physical Building Blocks that Support Concurrent Construction and Frictionless Interaction. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA, Article 26, 12 pages. https://doi.org/10.1145/3526113.3545658

[40] Danny Leen, Raf Ramakers, and Kris Luyten. 2017. StrutModeling: A Low-Fidelity Construction Kit to Iteratively Model, Test, and Adapt 3D Objects. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. Association for Computing Machinery, New York, NY, USA, 471–479. https://doi.org/10.1145/3126594.3126643

[41] Javier Marco, Ian Oakley, Eva Cerezo, and Sandra Baldassarri. 2013. Designing and Making a Tangible Tabletop Game with ToyVision. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction* (Barcelona, Spain) *(TEI '13)*. Association for Computing Machinery, New York, NY, USA, 423–426. https://doi.org/10.1145/2460625.2460719

[42] Nicolai Marquardt and Saul Greenberg. 2007. Distributed Physical Interfaces with Shared Phidgets. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction* (Baton Rouge, Louisiana) *(TEI '07)*. Association for Computing Machinery, New York, NY, USA, 13–20. https://doi.org/10.1145/1226969.1226973

[43] Michael Nebeling. 2017. Playing the Tricky Game of Toolkits Research. In *workshop on HCI. Tools at CHI*.

[44] Uwe Oestermeier, Philipp Mock, Jörg Edelmann, and Peter Gerjets. 2015. LEGO Music: Learning Composition with Bricks. In *Proceedings of the 14th International Conference on Interaction Design and Children* (Boston, Massachusetts) *(IDC '15)*. Association for Computing Machinery, New York, NY, USA, 283–286. https://doi.org/10.1145/2771839.2771897

[45] Antti Oulasvirta and Kasper Hornbæk. 2016. HCI research as problem-solving. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 4956–4967.

[46] Satyendra C Pandey and Srilata Patnaik. 2014. Establishing reliability and validity in qualitative inquiry: A critical examination. *Jharkhand journal of development and management studies* 12, 1 (2014), 5743–5753.

[47] Daniela Petrelli, Luigina Ciolfi, and Gabriela Avram. 2023. Envisioning, designing, and rapid prototyping heritage installations with a tangible interaction toolkit. *Human–Computer Interaction* 38, 2 (March 2023), 118–158. https://doi.org/10.1080/07370024.2021.1946398 Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/07370024.2021.1946398.

[48] C. Pike. 2002. Exploring the Conceptual Space of LEGO: Teaching and Learning the Psychology of Creativity. *Psychology Learning & Teaching* 2 (2002), 87 – 94. https://doi.org/10.2304/plat.2002.2.2.87

[49] Laura Pruszko, Hongri Gu, Julien Bourgeois, Yann Laurillau, and Céline Coutrix. 2023. Modular Tangible User Interfaces: Impact of Module Shape and Bonding Strength on Interaction. In *Proceedings of the Seventeenth International Conference on Tangible, Embedded, and Embodied Interaction* (Warsaw, Poland) *(TEI '23)*. Association for Computing Machinery, New York, NY, USA, Article 1, 15 pages. https://doi.org/10.1145/3569009.3572731

[50] Rebrickable®. [n. d.]. What is RebrickNet?

[51] Ben Redwood, Filemon Schffer, and Brian Garret. 2017. *The 3D printing handbook: technologies, design and applications.* 3D Hubs.

[52] Calvin Rubens, Sean Braley, Julie Torpegaard, Nicklas Lind, Roel Vertegaal, and Timothy Merritt. 2020. Flying LEGO Bricks: Observations of Children Constructing and Playing with Programmable Matter. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction* (Sydney NSW, Australia) *(TEI '20)*. Association for Computing Machinery, New York, NY, USA, 193–205. https://doi.org/10.1145/3374920.3374948

[53] Munehiko Sato, Ivan Poupyrev, and Chris Harrison. 2012. Touché: enhancing touch interaction on humans, screens, liquids, and everyday objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 483–492.

[54] Valkyrie Savage, Colin Chang, and Björn Hartmann. 2013. Sauron: Embedded Single-Camera Sensing of Printed Physical User Interfaces. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 447–456. https://doi.org/10.1145/2501988.2501992

[55] Denise Schmandt-Besserat. 1979. Reckoning before writing. *Archaeology New York, NY* 32, 3 (1979), 22–31.

[56] Evropi Stefanidi, Julia Dominiak, Marit Bentvelzen, Paweł W. Woźniak, Johannes Schöning, Yvonne Rogers, and Jasmin Niess. 2023. MagiBricks: Fostering Intergenerational Connectedness in Distributed Play with Smart Toy Bricks. In *Proceedings of the 22nd Annual ACM Interaction Design and Children Conference* (Chicago, IL, USA) *(IDC '23)*. Association for Computing Machinery, New York, NY, USA, 239–252. https://doi.org/10.1145/3585088.3589390

[57] Benedict Steuerlein and Sven Mayer. 2022. Conductive Fiducial Tangibles for Everyone: A Data Simulation-Based Toolkit using Deep Learning. *Proceedings of the ACM on Human-Computer Interaction* 6, MHCI (Sept. 2022), 183:1–183:22. https://doi.org/10.1145/3546718

[58] Vitamin Studio. 2012. Bit Planner. http://www.bit-planner.com Last accessed 2 September 2023.

[59] Saraha Ueno, Kunihiro Kato, and Homei Miyashita. 2016. A Tangible Interface to Realize Touch Operations on the Face of a Physical Object. In *Adjunct Proceedings of the 29th Annual ACM Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 81–83. https://doi.org/10.1145/2984751.2985711

[60] Joel Vidal Verdaguer, Guillem Vallicrosa Massaguer, Robert Martí Marly, and Marc Barnada. 2023. Brickognize: Applying Photo-Realistic Image Synthesis for Lego Bricks Recognition with Limited Data. *Sensors, 2023, vol. 23, núm. 4, p. 1898* (2023).

[61] Ana Villanueva, Zhengzhe Zhu, Ziyi Liu, Feiyang Wang, Subramanian Chidambaram, and Karthik Ramani. 2022. ColabAR: A Toolkit for Remote Collaboration in Tangible Augmented Reality Laboratories. *Proceedings of the ACM on Human-Computer Interaction* 6, CSCW1 (April 2022), 81:1–81:22. https://doi.org/10.1145/3512928

[62] Nicolas Villar, Florian Block, Dave Molyneaux, and Hans Gellersen. 2006. VoodooIO. In *ACM SIGGRAPH 2006 Emerging Technologies* (Boston, Massachusetts) *(SIGGRAPH '06)*. Association for Computing Machinery, New York, NY, USA, 36–es. https://doi.org/10.1145/1179133.1179170

[63] Nicolas Villar, Daniel Cletheroe, Greg Saul, Christian Holz, Tim Regan, Oscar Salandin, Misha Sra, Hui-Shyong Yeo, William Field, and Haiyan Zhang. 2018.

Project Zanzibar: A Portable and Flexible Tangible Interaction Platform. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (, Montreal QC, Canada,) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3174089

[64] Nicolas Villar, Kiel Mark Gilleade, Devina Ramduny-Ellis, and Hans Gellersen. 2006. The VoodooIO gaming kit: a real-time adaptable gaming controller. In *Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology* (Hollywood, California, USA) *(ACE '06)*. Association for Computing Machinery, New York, NY, USA, 1–es. https://doi.org/10.1145/1178823.1178825

[65] M Wilson. 2012. How GM is saving cash using legos as a data viz tool. *Fast Company* (2012).

[66] Hei Wong and Hiroshi Iwai. 2005. The road to miniaturization. *Physics World* 18, 9 (2005), 40.

[67] Peta Wyeth. 2008. How Young Children Learn to Program With Sensor, Action, and Logic Blocks. *Journal of the Learning Sciences* 17, 4 (2008), 517–550. https://doi.org/10.1080/10508400802395069 arXiv:https://doi.org/10.1080/10508400802395069

[68] Jishuo Yang and Daniel Wigdor. 2014. Panelrama: Enabling Easy Specification of Cross-Device Web Applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 2783–2792. https://doi.org/10.1145/2556288.2557199

[69] Arika Yoshida, Buntarou Shizuki, and Jiro Tanaka. 2015. Capacitive Blocks: A Block System That Connects the Physical with the Virtual Using Changes of Capacitance. In *Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Daegu, Kyungpook, Republic of Korea) *(UIST '15 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 85–86. https://doi.org/10.1145/2815585.2815731

[70] 2023 © Brickit Inc. [n. d.]. Build new things from your good old bricks.

A TOOLKIT DEVELOPMENT PROCESS

Here, we present the successive iterations of ProtoBricks, as we believe that sharing the challenges and knowledge from the development process can streamline the replication or modification for others.

The final form of the toolkit required three iterations. Testing different electronic components, available 3D-printing materials, and techniques eventually resulted in meeting all specified requirements. To present the complete workflow, we describe two prototype versions that did not meet all of the specified requirements, yet were essential for the developement of the final version of the ProtoBricks toolkit.

A.1 First prototype—resistance change

The first method to detect the bricks was to use one of the basic features of the Arduino board—measuring the voltage using the pin with an analog-to-digital (A/D) converter. Our initial approach was to assign significantly different voltages to the 2x2 plates arranged as a matrix by implementing a voltage divider (see c) on Figure 13). The base PCB had the form of a potentiometer where the voltage distribution was controlled by the position of the brick acting as a voltmeter (see b) on Figure 13). Despite the high efficiency, simplicity of the base design, and the ability to detect many bricks on the board as well as upwards, this solution required connecting wires to the brick, impading usability.

A.2 Second prototype—closing the electrical circuit

The second idea for the prototype was to develop a matrix of "buttons" by implementing a checkerboard of dots (a single inset in the LEGO® system), consisting of digital pins and GND (digital pins are indicated in green, while GND is represented in gray in a) and b) illustrations in Figure 14). The prototype assumed a field

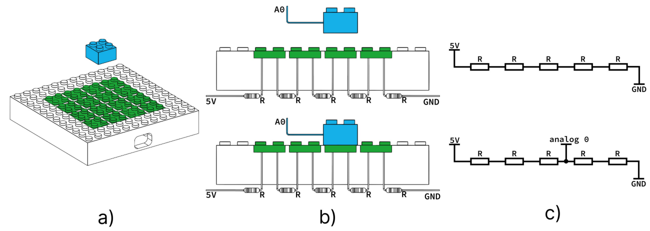

Figure 13: Functional diagram of the first version of the toolkit based on the analog reading of the voltage value on the 2x2 plate.

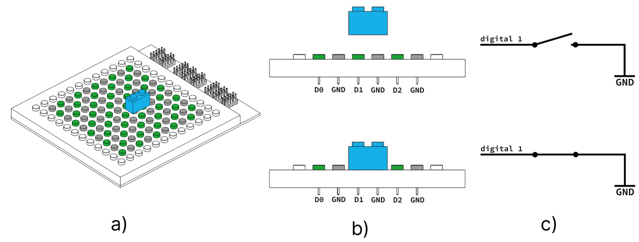

Figure 14: Functional diagram of the second version of the prototype based on the detection of closing the circuit by a conductive brick.

size of 10 x 10 dots, which required connecting 50 digital pins and a common ground to the Arduino MEGA (see a) inFigure 14). In order to simplify the connection, we designed a dedicated PCB for easy assembly. The PLA base was screwed to the printed circuit, and then the dots were glued and heat-set inserted with a wire so that it was possible to create a connection between the print and the electronic layer of the system. The "button" circuit was closed by placing the brick on the surface and connecting the digital pins declared in the program as INPUT-PULLUP to GND, which was detected as a change state from HIGH to LOW (see b) and c) illustrations in Figure 14). In this case, it was possible to use bricks of different sizes (starting from 1x2) and orientations on the entire surface of the base. Moreover, the implementation of the detection pins (digital pins) and the detected factor (GND) inside the main board did not require to connect the bricks to the wire. However, the disadvantage of this approach was the possibility of detecting only the first layer of bricks, which prevented recognizing higher structures (bricks stacked on top of each other). Another technical problem was the fragility of individual dots in the base. Due to the technique of making 3D filament conductive by adding carbon, the material has less cohesion between layers and is more brittle. As a result, the pins were damaged during the tests, making it impossible to use a fragment of the board.

A.3 Final prototype—capacitance change by touch

As the results obtained from the second prototype indicated susceptibility to breakage and fragility when using individual pins, we have decided to return to the 2x2 sensor plate configuration on the base (see c) on Figure 15). Two types of 3D material were used in each iteration. The main filament used in ProtoBricks development was PLA due to its availability, accessibility, and a large palette of colors, allowing for versatile combinations. Additionally, we used conductive filament as a carrier of sent and received signals. Previous works utilizing conductive 3D printing to measure electrical capacity focused on exploring human touch as a trigger [27, 32]. However, none of these endeavours explored this technique to detect tangible bricks in 3D space. Another system was Capacitive Blocks [69], where the bricks were equipped with two shells of conductive material to form a capacitor. However, the disadvantage of this solution was the limitation of the possibility of precise recognition of the number of bricks. Moreover, the detected capacity was characterized by significant issues with the consistency of values and the lack of repeatability of readings, especially for stacks of more than two bricks. Despite the use of the dimensions of the LEGO® 1x1 brick, the connection between the inner and outer of the capacitor shell with the 3-pin system inside the base was not optimal. Our solution is based on detecting the brick's position by detecting the high capacitance of our hand applied to the brick, which becomes the extender of the electrode located in the base. This is enabled by the MPR121[17], which is a capacitance sensor mainly focused on touch detection (see Figure 15). The sensor transmits its reading via I2C communication to the Arduino UNO board, where the values are appropriately converted to a position by applying a mask between the sensor pin numbers and the coordinates connected plates in the base matrix. The connection between the base plate and the rest of the electronics was designed in a similar way as in the second prototype—by designing and making a PCB that allows connection with the printout through a heat-set insert. In order to simplify the project, a coherent I2C pin has also been added on the board, and sockets for MPR121 headers along with the implemented address change connection required to perform when using two or more touch sensors. Due to the high capacitance value of our body [53], it is possible to detect bricks that are placed in very high stacks. Due to the sensor's readings, primarily influenced by the capacitance of our body, the mechanism requires placing one brick at a time. Consequently, the placement of two or more bricks will be detected as a single action of adding one brick. Height detection is based on a software analysis of the detected number of bricks in one position and incrementing the variable assigned to a specific plate. During testing, we reached a height of 30 bricks. Higher values were difficult to achieve due to the stability of a single tower caused by the 3D technology used, not due to the capacity measurement technology used.

During the development of the prototype, tests were also carried out on the possibility of detecting the parasitic capacitance of the bricks themselves without the touch—hand factor, but the value was insufficient to clearly determine the operation of adding the brick. Our bricks are too small for this technique compared to other research works [5, 26] where this has been used. We also experimented with larger dimensions for the parts (200% of bricks inspired by DUPLO™ series), which ended with a positive result. Another solution would be to use a dedicated capacitance sensor sensitive to small changes. Unfortunately, both ideas did not meet the assumptions of compatibility with the LEGO® System and ease of assembly for people not specialized in electronics.

As an integral part of the prototype, an Adafruit TCS34725 color sensor[18] has been implemented to enhance functionality. Placed within a specially designed slot, calibrated to fit the size of the single brick, the sensor scans the color of the inserted brick. Employing I2C communication, it then transmits RGB pigment information to the Arduino. Subsequently, the Arduino decodes this data, translating it into the corresponding brick colors.

A.4 Dimensions and shape of bricks and dots

In parallel with the tests of various prototypes, we conducted size and shape tests for the best fit with both 3D-printed bricks and the LEGO® System. The printouts that were adapted in terms of dimensions and design consist of the bricks, the conductive plate of the base, and a color frame that is both a holder of conductive 2x2 plates and a mounting element with the rest of the case.

As a starting point, we used the base dimensions of LEGO® 2x2 brick—15.8 x 15.8 mm in width and length, a height of 9.6 mm of the base, and a dot diameter of 4.8 mm with 1.7mm of height. Due to the use of the FFF printing method, there is a difference in size between the designed model in CAD and the actual printout. In order to adjust the dimensions, it is possible to use tables of rules with recommended values useful to be considered in the design [51]. However, they are a general presentation not based on a specific printer model, material, or g-code settings. The list of parameters affecting the change of dimensions is the precision of the printer's stepper motors, belt tension, nozzle size, temperature and flow material, and layer thickness, as well as the model itself. Due to all these differences, we decided to always print on the same printer—the two-extruder Raise3D E2[19]. During six sessions of iterating dimensions, we determined that reducing the brick dimensions to 98% would be the most optimal, with the exception of the height of the dot, which is 22% higher, in order to increase the contact area of the conductive printouts, while maintaining the ease of their separation. However, in the case of conductive base plates and base frames, which are made of one material and did not require printing with supports, the most optimal dimension was 99%.

While designing the shape of the brick, we started with a model that would provide the largest contact surface, fully adhering to both the dots and the base of the brick (see a) in Figure 16). However, this solution required full, perfect fit and uniformity between prints, which made the process of achieving easy replication difficult. In addition, after obtaining a perfect fit, the lack of free space caused difficulty separating and fast abrasion of the surface, resulting in failure to connect. The following attempts were based on a classic shape with a central, round tab ensuring good connection while allowing the bricks to fit together (see b), c) and d) in Figure 16). With this method, the connection of the bricks met all the criteria. However, it was necessary to cover the inner surface of the brick with conductive PLA in addition to the ring in order to ensure a good electrical connection (see c) of Figure 16). Because of this aspect, it was necessary to specify the thickness of both colored and conductive walls. Using the most popular 0.4 mm nozzle, we

[17]https://www.adafruit.com/product/1982

[18]https://www.adafruit.com/product/1334

[19]https://www.raise3d.com/products/e2/

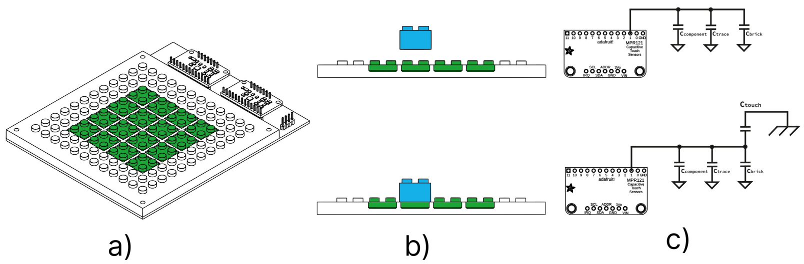

Figure 15: Functional diagram of the final version of the prototype based on a capacitive sensor triggered by the body capacity value transmitted by the brick.

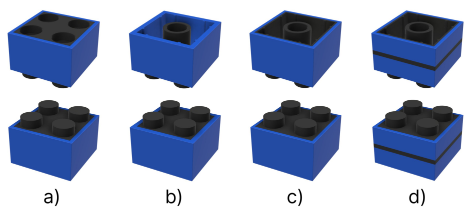

Figure 16: The evolution of our smart bricks compatible with the capacitive detection system implemented inside the base.

had three wall lines—2 colored PLA and one conductive. In addition, we used a small gap between them of 0.05 mm to prevent excessive mixing of materials, thus increasing the resistance of the pads and darkening the color of the shell. The last iteration was based on tests using the prototype and was related to different habits of catching bricks also without contact with the upper surface. In order to ensure electrical continuity between the base and the finger, regardless of the place of touch, a 1mm strip of conductive paint, and in the final iteration of conductive PLA, has been added around the brick to detect contact with any surface of the model (see d) on Figure 16).

B SUMMARY OF PROTOBRICK CONSTRUCTIONS AT WORKSHOPS

Group 1

Physicalization The first group divided the 30 data points of daily energy consumption into five parts, each represented by a tower placed on one of five plates. Each tower was constructed from bricks, representing the average energy usage over six days. The values were assigned colors ranging across the spectrum from blue to red. Each bar consists of the sum of the values assigned to the colors of the blocks, representing the average kWh used (see left on Figure 17).

Rapid prototyping In the second task, the group assigned two plates on a matrix for sound control and two lamps. On the first plate, placing a yellow brick activated the output, while a red brick deactivated it. The second plate featured a tower whose height controlled the brightness of the light or the volume of the sound. This was demonstrated using two LEDs and a buzzer. The final functionality involved controlling the degree of window opening, represented by the height of a tower made from blue bricks on a matrix plate, which corresponded to the position of a servo mechanism (see right on Figure 17).

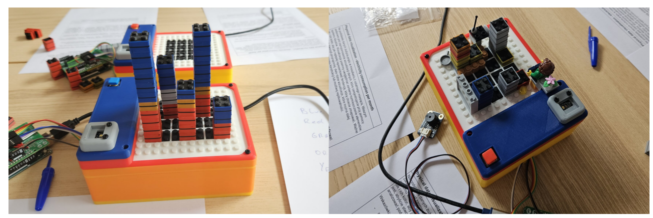

Figure 17: Construction of the first group for physicalization and rapid prototyping during workshops

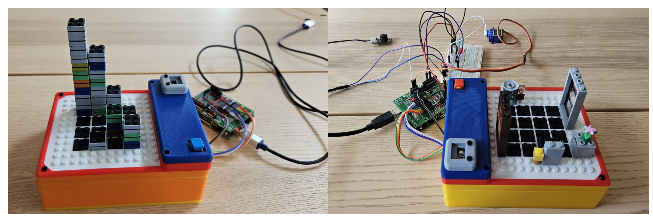

Figure 18: Construction of the second group for physicalization and rapid prototyping during workshops

Group 2

Physicalization The second group concentrated on visualizing the peak hour of highest power usage in kilowatts (kW) during the day. They utilized twelve of the sixteen plates on the matrix, assigning each plate a 2-hour slot from divided the 24-hour day. Each day was represented by a brick, which varied in color from gray to red, corresponding to the range of peak values. The height of one tower represented the amount of power used during a two-hour slot, while the color of the brick indicated the peak value (see left on Figure 18).

Rapid prototyping In the rapid prototyping task, they assigned one plate on the matrix to each functionality and applied a uniform scale for all components where blue indicated "off." Each subsequent color in the four-color spectrum from green to red indicated increasing intensity of light, degree of window opening, or sound frequency (see right on Figure 18).

Group 3

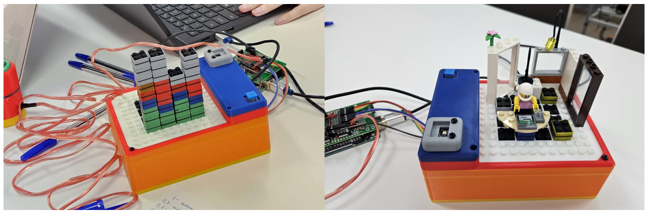

Figure 19: Construction of the third group for physicalization and rapid prototyping during workshops

Physicalization The last group decided to represent data from four parts - the average energy consumption in kWh per week, represented by individual towers arranged in a single line. Unit values were assigned to each of the five brick colors. Each column consists of the sum of values assigned to the colors of the bricks, representing the average kWh consumed (see left on Figure 19).

Rapid prototyping In the next task, instead of adding blocks like previous groups, the team utilized touch detection. Each of the four functionalities was assigned to one plate on the array and paired with a corresponding color. Two light points were assigned one yellow brick each, opening a window got a gray brick, and activating sound a blue brick. These bricks serve as switches that toggle the state to the opposite, turning on or off an electronically connected Arduino component—LED, buzzer, or servo (see right on Figure 19).