



## Tangibilisation of an educational scheduling game's graphical interface

Benoît Landi, Catherine Pons Lelardeux, Pierre Lagarrigue, Jean-Pierre Jessel

### ► To cite this version:

Benoît Landi, Catherine Pons Lelardeux, Pierre Lagarrigue, Jean-Pierre Jessel. Tangibilisation of an educational scheduling game's graphical interface. 32ème Conférence francophone sur l'Interaction Humain-Machine (IHM 2021), Apr 2021, Virtual Event, France. pp.1-5, 10.1145/3451148.3458637 . hal-03562159

**HAL Id: hal-03562159**

**<https://hal.science/hal-03562159>**

Submitted on 8 Feb 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Tangibilisation of an educational scheduling game's graphical interface

Tangibilisation de l'interface d'un jeu de formation à la planification

Benoît Landi

benoit.landi@univ-jfc.fr

IRIT, University of Toulouse, INU Champollion, Serious  
Game Research Lab  
Albi, France

Pierre Lagarrigue

pierre.lagarrigue@univ-jfc.fr

ICA, University of Toulouse, INU Champollion, Serious  
Game Research Lab  
Albi, France

Catherine Pons Lelardeux

catherine.leardeux@univ-jfc.fr

IRIT, University of Toulouse, INU Champollion, Serious  
Game Research Lab  
Albi, France

Jean-Pierre Jessel

Jean-Pierre.Jessel@irit.fr

IRIT, University of Toulouse, Université Toulouse III - Paul  
Sabatier  
Toulouse, France

## ABSTRACT

TUI or Tangible User Interface is a type of interfaces that merges the physical and digital. They are usually opposed to Graphical User Interface, or GUI, where the physical devices (e.g. keyboard and mouse) and the digital objects displayed (e.g. mouse cursor, buttons, ...) are distinct. According to the state of the art, TUIs offers a lot of promised advantages. Hence bringing a GUI into the tangible could be beneficial to the players' outcomes. This paper presents such conversion, using a nursing students training game, and the choices made during such design. *Clone*, the game, teaches scheduling to nursing students in France. The TUI recreates *Clone*'s scheduling interface using 3d printed tokens and board and connects those object to the game using image recognition and pattern on the bottom of the 3d printed elements.

## CCS CONCEPTS

• **Human-centered computing** → **Empirical studies in interaction design**; • **Applied computing** → **Interactive learning environments**.

## KEYWORDS

Tangible, TUI, Formation, Scheduling, Nurse

## RÉSUMÉ

Les TUIs ou *Tangible User Interfaces* (Interface utilisateur tangibles) sont un type d'interface qui combine physique et numérique. Elles sont généralement opposées aux *Graphical User Interfaces*, ou GUI (Interface utilisateur graphique) où les éléments physiques (c.à.d. clavier/souris) et les éléments numériques affichés (c.à.d. curseur de souris, boutons,...) sont séparés. En regardant la littérature, les

TUIs sembleraient offrir un grand nombre de bénéfices. Convertir une GUI en TUI pourrait alors améliorer les résultats utilisateurs. Ce papier présente une telle conversion d'un jeu éducatif pour les élèves infirmiers, ainsi que les choix effectués pendant cette conversion. *Clone*, le jeu, enseigne la planification à des élèves infirmiers français. La nouvelle TUI recrée la partie planification de *Clone* en utilisant des objets et un terrain de jeu imprimés en 3D et connecte ces pièces au jeu en utilisant de la reconnaissance d'image et des symboles distinctifs placés sous les objets.

## MOTS-CLÉS

Tangible, TUI, Formation, Planification, Infirmier

### ACM Reference Format:

Benoît Landi, Catherine Pons Lelardeux, Pierre Lagarrigue, and Jean-Pierre Jessel. 2021. Tangibilisation of an educational scheduling game's graphical interface : Tangibilisation de l'interface d'un jeu de formation à la planification. In *32e Conférence Francophone sur l'Interaction Homme-Machine (IHM '21 Adjunct)*, April 13–16, 2021, Virtual Event, France. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3451148.3458637>

## 1 INTRODUCTION

### 1.1 Tangible User Interfaces

The name Tangible User interface (TUI) is used to describe a broad range of interfaces which let users interact through physical objects [4]. TUIs, originally called Tangible bits, were formally introduced by Brygg Ullmer and Hiroshi Ishii in 1997 [10], along with a lot of different perspectives to what could be done with them. For example an interactive information grid [11] or information bricks [5]. According to Hornecker et al.[9], while TUIs encompass a large spectrum of interfaces, they 'share the following characteristics: tangibility and materiality, physical embodiment of data, embodied interaction and bodily movement as an essential part of interaction, and embeddedness in real space'. Tangible User Interfaces (TUIs) are usually defined in opposition to the more standard Graphical User Interfaces (GUIs). GUIs' user interacts through physical input devices (e.g. mouse and keyboard) on a entirely digital interface. GUIs are relatively indirect and loosely coupled [14] whereas TUIs

provide a much closer coupling between the physical and the digital – to the extent that the distinction between input and output becomes increasingly blurred’[14]. This paper presents the conversion (tangibilisation) of a digital interface into a token+constraint [21] tangible interface. Ultimately, our goal is to study the impact (usability, motivation,...) of this tangibilisation and see whether we can get benefits shown in other studies [6, 8, 11, 15, 20]. This paper focuses on the design choices made to recreate the GUI as a TUI.

## 2 INTERFACES

### 2.1 Original GUI : A nursing care’s scheduling training game

Clone is a simulation game dedicated to scheduling training for nursing students. In its original GUI form, the user interacts using the mouse. Clone’s gameplay centres around two activities: scheduling and executing. For the conversion, we decided to focus only on the scheduling part as the executing isn’t well suited for such conversion as it is a much more complex gameplay. The scheduling’s GUI can be seen in fig. 1.

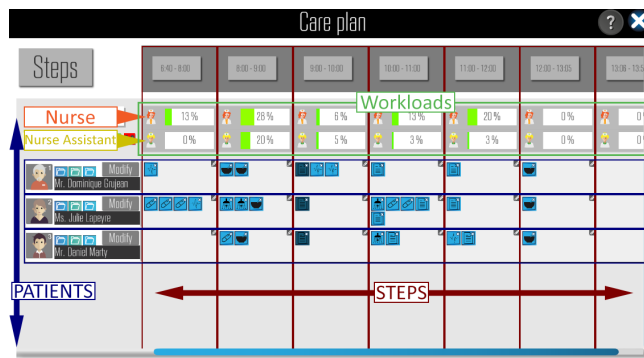


Figure 1: Clone schedule’s GUI with Patients (Blue lines), Steps (Red columns). On top of the schedule are the workloads (green) for each steps; each time, one for the Nurse (orange) and the Assistant (yellow).

The daily calendar (fig. 1) is split into steps (red columns), each representing a stage in the typical daily nursing care and patient (blue lines). As feedback there are two total workloads for each steps, one for the nurse’s activities for that step (redhead person, top one) and one for the nurse’s assistant’s activities for that step (blond person, bottom one).

To schedule a new activity, the user uses a sub-interface (fig. 2). This sub-interface focuses on one of the slots (intersection of a patient’s line and a step’s column). In this sub-interface, activities can be selected from the left (blue) and dropped in the selected step (red middle). The workload (green) is also shown with only the selected patient workload on the left and the whole step workload on the right. Activity delegation can be done for the activities that allow it by drag and dropping the actor from the delegation list (yellow) on an already schedule activity.

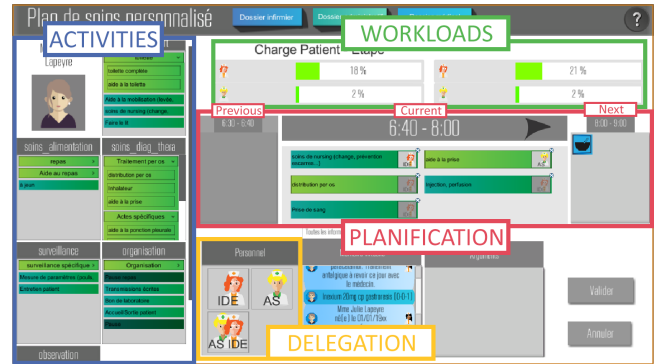


Figure 2: Individual care plan. : Clone’s sub-interface where the user schedules and delegates activities for one step and one patient

### 2.2 GUI to TUI

The tangible object is designed to stay as close as possible to the digital version (fig. 3) but using a modular-board and tokens, all 3D-printed. The information is organised in the same way with lines for patients and columns for steps. One absent feedback is the workload (Nurse and Nurse Assistant in fig. 1) which is not directly visible on the physical object but will still be updated on the screen in real time. Instead of a sub-interface (fig. 2) to drag and drop activities in the GUI, in the TUI the user pick activities tokens and place them on the board.

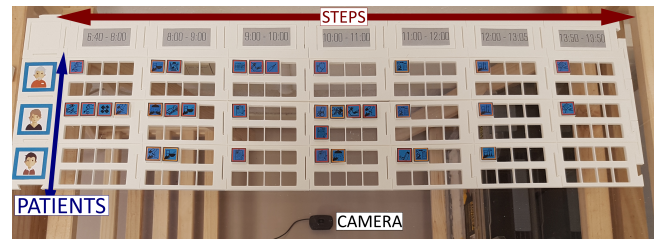
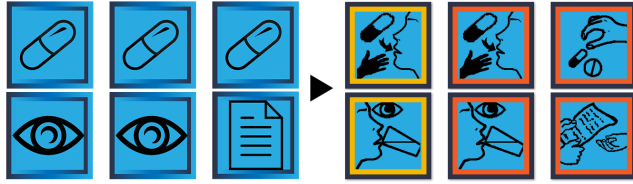


Figure 3: The TUI design with the patients images (first column), the time steps (the columns), the activities (tokens at the intersections of the patients and time steps)

### 2.3 Activity tokens’ symbols

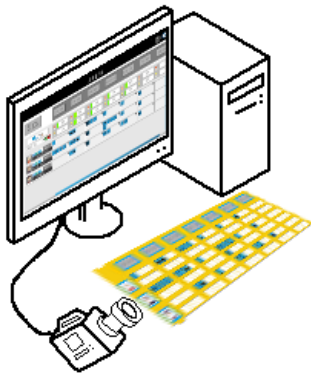
In order to stay as similar as possible to the original, we forced ourselves to make only the absolute necessary changes. One such change was to redesign the icons on the activities’ tokens. In the GUI interface the token uses symbols that reflect the category of the activity which means that different activities have the same symbol (see fig. 1). In the GUI, the user could always get the activity name by hovering the icon. This is not possible with our TUI’s object. Therefore, we designed, with the help of one of the nurse teacher involved in a Nursing School, a new icon for each activity (see fig. 4). Those new TUI’s icons will then be implemented in the GUI. A similar issue arose with activities’ delegation (to the assistant) which wasn’t directly visible on the activity icon. In addition to

new icons, we also added a coloured border to symbolise the actor tasked with the activity (orange/red for the user's character and yellow for the assistant).



**Figure 4: Original GUI's Tokens (left) and new GUI and TUI's tokens (right). Borders indicate the delegation : orange for the nurse and yellow for the nurse assistant**

The results can be seen with some examples in fig. 4. Top-left are both "Help with oral pill taking" yellow border for the nurse assistant and orange for the nurse. Similarly bottom-left are both "Hydration Surveillance", again with yellow border for the nurse assistant and orange for the nurse. And on the right, two activities that can't be delegated to the nurse assistant : top "Distributing oral pills" and bottom "Written reports transmission".



**Figure 5: The tangible object is connected to the software using a camera that reads pattern on the back of each token**

### 3 THE TANGIBLE INTERFACE INNER-WORKINGS

#### 3.1 Technological solution

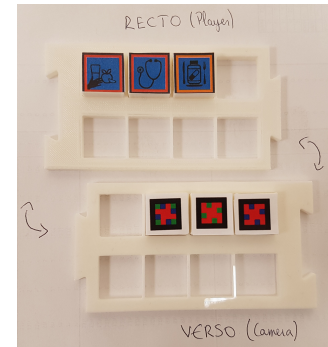
**3.1.1 Techniques.** The link between system based on TUIs and software can be made in a variety of ways. When choosing a solution we used the following criterion: object size, total cost, production time, durability, transportability and user-friendliness. RFID chips (see [7, 16, 19]) are quite durable and not too hard to transport but they make prototype expensive with a lot of tokens and big as their receivers' ranges need to not overlap. Integrated circuits and printed ones don't have RFID's size issue however they are far less durable and are still expensive (per token). The solution we chose was image detection. Image detection allows us to keep it cheap (the only costly part would be the camera), easy to update and quite

durable (all the parts are 3D printed with some paper symbols) and transportable (see fig. 5).

**3.1.2 Patterns Detection software/libraries.** For analysing the video feed, there is multiple possible software, a lot of them focus on Augmented Reality (AR). As an illustration:

- ReacTIVision [12] an open-source framework design for table-based games [2, 22].
- OpenCv[1] is a library of functions focused around computer vision and image computing [13, 23]
- ARTag a computer vision software for AR using 2D markers [3]

Such pattern recognition can be done in two ways: (1) by positioning a camera above the objects and detecting the patterns from above [17], (2) by positioning a camera below the objects and detecting the bottom pattern of the objects [22]. We decided to go for the camera below the pattern (see fig. 6). It allowed us to hide the technical parts for the user by putting the patterns below (or not have to use a projector). This, however, meant that we would need a transparent table for the camera to see the bottom of the object. We went for a Plexiglas board with some sawhorses making it easier for transport. We ended up with a 130x90 cm plexiglass table and a 75.5x83 cm object with 14 patients, the maximum.



**Figure 6: Both faces of a slot with 3 activities. The Recto (Top) is what the user see while the Verso (Bottom) is what the camera detect.**

#### 3.2 Patterns

When we designed our system, our main concern was to : have activity token be small as to avoid having an overall object too big (more than 1m in length) and have enough variability in the patterns to hold all possible activities. After some evolutions (see fig. 7) we ended up with the pattern seen in fig. 8. This pattern is the results of multiple choices we made to answer multiple issue we had.

- We first shifted from the QRCode to personalised pattern as regular QR would have been too big for our use (we did not find any libraries usable with openCV for MicroQR).
- We chose to concentrate the data by using colours instead of the 'simple' black and white binary. We chose 6 colours (Red, Green, Blue, Magenta, Cyan and Yellow) to have colours as different as possible.



- Our initial idea was to use pattern matching to identify the elements in our video feed. Our idea was to first detect the border then identify the content. Hence we started with a checkered border that would be very specific and so identifiable in our image.
- In one of our first optimisations, we use contour findings to find the marker candidates. This meant that we needed a line border.
- Our main evolution was to remove the pattern matching for the identification and use a method that is more adapted to our simple patterns : image cutting (and used in the Aruco library). The idea is, knowing the numbers of 'bits' in the pattern, you cut average the colours in the bits' area to get the colour and that gives you the pattern values directly. This means that it is better to use larger borders to have every element of the same size and it also reduces the number of contours in the image and reduces false positives in marker candidates.
- Finally, after realising that we would need a smaller than anticipated number of different patterns, we reduced our colours from 6 to 3. This made the colour identification simpler as the software would not confuse some colours (mainly yellow to green when it was not well lit).

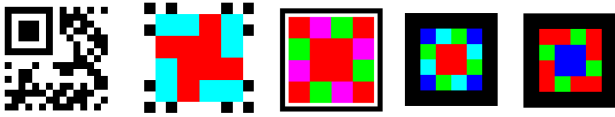


Figure 7: The evolution of our pattern from QRCode to the coloured version, from checkered border to line to heavy line and from 6 to 3 colours

The final patterns (see fig. 8) are coded using coloured 'bits'. Each of this 'bits' is repeated to make it identical regardless of the pattern's orientation. The corresponding 'bits' also provide some error correction as the detection returns the most common colour among them. To get the pattern's value/id, we use a ternary system, much like a white (0) and black (1) 'bits' would use binary. Here we have red(0), green(1) and blue(2). In fig. 8, you can see the an activity pattern and a step pattern. The patients' pattern are similar to steps' patterns with the central bit being red instead of blue.

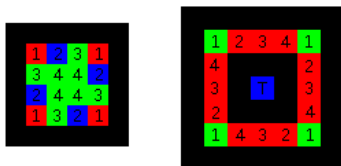


Figure 8: Activity (left) and Steps (right) Patterns with corresponding 'bits' numbered (1 to 4) and type for steps' pattern

### 3.3 Detection

**3.3.1 Detection Loop.** Our detection loop for each image from the video feed uses the following steps :

- **All contour finding** : lists all the contour using opencv's function [18]
- **Reference size** : finds the triangle pattern to get the reference size (and pattern-to-reference ratio)
- **Marker candidate sorting** : uses the ratio and other indicator to filter the contour and gets the likely markers
- **Squaring the candidates** : reorders the points of the candidate
- **Project to square** : projects the candidate to a square shape
- **Marker creation** : cuts the marker image into its coloured bits' to get its id (and check its validity)

This info is then passed to the Unity game and used to update the in-game schedule.

**3.3.2 Size Calibration.** In order to automatically deal with the possible difference in size between the video input and the patterns used as reference, the first element searched is the size calibration triangle. The idea is to use an element which is quite different in size and shape to the others. Thus, it can be identified and measured with some certainty. We then use the ratio between the video feed's triangle and the reference's triangle to estimate the expected size of the patterns in the video feed.

**3.3.3 Pre-matching and looping matching.** We also reduce calculation by minimising the number of patterns we are looking for each loop. As the relative position of camera should not change during the experimentation, the image sizes in the video feed won't either. Therefore, we can split the elements to look for in two categories:

- static elements which should not change during the interactions such as user ID, step ID, channel and size calibration (triangle size).
- droppable elements which need to be continuously checked for any changes: the activity items.

While we need to update the droppable element as often as possible, we can be more lax with the static element. This allows us to use the position of those static elements and reduce the droppable elements' search area as much as possible.

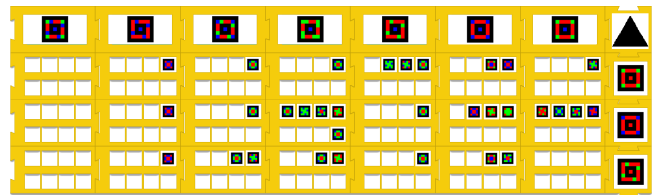


Figure 9: The backside of the prototype. The reference pattern is on the top-right (the triangle), the patients patterns on the right, the steps patterns on top and the activities patterns on the rest

## 4 PERSPECTIVES

With the TUI presented in this paper, we want to study the impacts of the TUI on the users. The experiments' population will be nursing students from the already existing course that uses the original Clone. The experiment uses different measures (multiple questionnaires and in-game metrics). By having the TUI be as close as possible to the original GUI, we hope that any difference on the measures, between TUI and GUI, be the impact of the tangibilisation. After that first experimentation, our secondary goal is to find metrics/aspect of special interest. We would make variants of the TUI/GUI pair, with only one aspect of interest modified. Such variants should give us insight in the relation of that aspect with the tangibility.

## REFERENCES

- [1] [n.d.]. *OpenCV*. <https://opencv.org/>
- [2] R. Bencina, M. Kaltenbrunner, and S. Jorda. 2005. Improved Topological Fiducial Tracking in the reacTIVision System. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, Vol. 3. IEEE, San Diego, CA, USA, 99–99. <https://doi.org/10.1109/CVPR.2005.475>
- [3] Mark Fiala. [n.d.]. ARTag Revision 1, A Fiducial Marker System Using Digital Techniques. ([n.d.]), 49.
- [4] Kenneth P. Fishkin. 2004. A taxonomy for and analysis of tangible interfaces. *Personal and Ubiquitous Computing* 8, 5 (Sept. 2004). <https://doi.org/10.1007/s00779-004-0297-4>
- [5] George Fitzmaurice, Hiroshi Ishii, and William Buxton. 2002. Bricks: Laying the Foundations for Graspable User Interfaces. (April 2002). <https://doi.org/10.1145/223904.223964>
- [6] George W. Fitzmaurice and William Buxton. 1997. An empirical evaluation of graspable user interfaces: towards specialized, space-multiplexed input. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*. ACM, Atlanta Georgia USA, 43–50. <https://doi.org/10.1145/258549.258578>
- [7] Pilar Castro Garrido, Guillermo Matas Miraz, Irene Luque Ruiz, and Miguel Angel Gomez-Nieto. 2011. Use of NFC-based Pervasive Games for Encouraging Learning and Student Motivation. In *2011 Third International Workshop on Near Field Communication*. IEEE, Hagenberg, Austria, 32–37. <https://doi.org/10.1109/NFC.2011.13>
- [8] Michael S. Horn, Erin Treacy Solovey, R. Jordan Crouser, and Robert J.K. Jacob. 2009. Comparing the use of tangible and graphical programming languages for informal science education. In *Proceedings of the 27th international conference on Human factors in computing systems - CHI '09*. ACM Press, Boston, MA, USA, 975. <https://doi.org/10.1145/1518701.1518851>
- [9] Eva Hornecker and Jacob Buur. 2006. Getting a Grip on Tangible Interaction: A Framework on Physical Space and Social Interaction. *CHI '06: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (April 2006), 10. <https://doi.org/10.1145/1124772.1124838>
- [10] Hiroshi Ishii and Brygg Ullmer. 1997. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '97*. ACM Press, Atlanta, Georgia, United States, 234–241. <https://doi.org/10.1145/258549.258715>
- [11] Robert J K Jacob, Hiroshi Ishii, Gian Pangaro, and James Patten. 2002. A Tangible Interface for Organizing Information Using a Grid. *CHI '02: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (April 2002), 8. <https://doi.org/10.1145/503376.503437>
- [12] Martin Kaltenbrunner and Ross Bencina. 2007. reacTIVision: a computer-vision framework for table-based tangible interaction. In *Proceedings of the 1st international conference on Tangible and embedded interaction - TEI '07*. ACM Press, Baton Rouge, Louisiana, 69. <https://doi.org/10.1145/1226969.1226983>
- [13] Wonwoo Lee, Woontack Woo, and Jongweon Lee. 2005. TARBoard: Tangible Augmented Reality System for Table-top Game Environment. (Jan. 2005), 5.
- [14] Claire O'Malley and Danae Fraser. 2004. Literature Review in Learning with Tangible Technologies. *NESTA Futurelab Rep.* 12 (Jan. 2004).
- [15] Mitchel Resnick, Fred Martin, Robert Berg, Rick Borovoy, Vanessa Colella, Kwin Kramer, and Brian Silverman. 1998. Digital manipulatives: new toys to think with. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '98*. ACM Press, Los Angeles, California, United States, 281–287. <https://doi.org/10.1145/274644.274684>
- [16] Mohamad Saffroin, Fahim Nur Cahya Bagar, and Fidelis Yoga Pralista. 2019. The Development of Digital Board Game to Introduce Indonesian Wildlife using AR Technology and NFC. In *2019 International Electronics Symposium (IES)*. IEEE, Surabaya, Indonesia, 277–282. <https://doi.org/10.1109/ELECSYM.2019.8901519>
- [17] Bertrand Schneider, Patrick Jermain, Guillaume Zufferey, and Pierre Dillenbourg. 2011. Benefits of a Tangible Interface for Collaborative Learning and Interaction. *IEEE Transactions on Learning Technologies* 4, 3 (July 2011), 222–232. <https://doi.org/10.1109/TLT.2010.36>
- [18] Satoshi Suzuki and Abe Keiichi. 1985. Topological Structural Analysis of Digitized Binary Images by Border Following. *Comput. Vis. Graph. Image Process.* 30 (1985), 15. [https://doi.org/10.1016/0734-189X\(85\)90016-7](https://doi.org/10.1016/0734-189X(85)90016-7)
- [19] Iván Sánchez, Marta Cortés, Jukka Riekk, and Mika Oja. 2011. NFC-based interactive learning environments for children. In *Proceedings of the 10th International Conference on Interaction Design and Children - IDC '11*. ACM Press, Ann Arbor, Michigan, 205–208. <https://doi.org/10.1145/1999030.1999062>
- [20] Philip Tuddenham, David Kirk, and Shahram Izadi. 2010. Graspables revisited: multi-touch vs. tangible input for tabletop displays in acquisition and manipulation tasks. In *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10*. ACM Press, Atlanta, Georgia, USA, 2223. <https://doi.org/10.1145/1753326.1753662>
- [21] Brygg Ullmer, Hiroshi Ishii, and Robert J. K. Jacob. 2005. Token+constraint systems for tangible interaction with digital information. *ACM Transactions on Computer-Human Interaction (TOCHI)* 12, 1 (March 2005), 81–118. <https://doi.org/10.1145/1057237.1057242>
- [22] Zhen Xie, Alissa Antle, and Nima Motamedi. 2008. Are tangibles more fun?: Comparing children's enjoyment and engagement using physical, graphical and tangible user interfaces. 191–198. <https://doi.org/10.1145/1347390.1347433>
- [23] Chen Zhao, Chih-Pin Hsiao, Nicholas M. Davis, and Ellen Yi-Leun Do. 2013. Tangible games for stroke rehabilitation with digital box and blocks test. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems on - CHI EA '13*. ACM Press, Paris, France, 523. <https://doi.org/10.1145/2468356.2468448>