POLITECNICO
MILANO 1863

Design and Technology Document

Advanced User Interfaces Course
A.Y. 2019/2020
Project "Turn Taking"

Smarta

Group 8:

Fabiana Ferrara
Stefano Formicola
Marta Mazzi
Elena Palombini

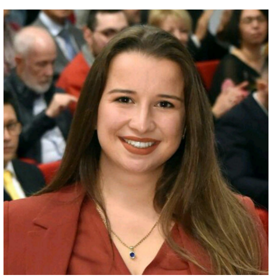
Abstract

This document presents SMARTA, a smart object in the shape of a ball, useful to manage conversations, mostly in the specific context of class discussions for students aged 5 to 14. The student who is holding the object is the only one allowed to talk. SMARTA helps each participant regulate the time they speak, by implementing a timer mechanism and signalling to the user, through light and vibration, when their turn to speak is about to end. The ball detects when it is passed to another user through an accelerometer and it resets the timer for a new turn. SMARTA also provides a web interface for the educator to set turn duration times and warn the students by making the ball emit red light when they speak over each other. The web interface also allows to consult statistics of past discussions. The core of SMARTA is a Raspberry Pi Zero, to which are connected a strip of LED lights and an accelerometer, as well as a vibration motor; the chosen programming language is Python. In this document we will propose also a method for empirical evaluation of our project, as well as indicate future directions that this work will take.

The team

Fabiana Ferrara
fabiana.ferrara@mail.polimi.it
+39 3803691440

Stefano Formicola
stefano.formicola@mail.polimi.it
+39 3348237110



Marta Mazzi
marta.mazzi@mail.polimi.it
+39 3315284247

Elena Palombini
elena.palombini@mail.polimi.it
+39 3273746937

Contents

1 Introduction

SMARTA is a smart object intended for teaching children, mainly but not only, how to entertain a conversation involving everybody, respecting the other participants and having fun at the same time. Whoever has the ball in their hands, is the one allowed to talk in that moment.
SMARTA is meant to be used during teaching activities that involve a group of kids aged between 5 and 14 years old, and a teacher who acts as supervisor. Even though it is mainly targeted to children in a school environment, this smart object can be used also in other kind of activities, for example team building with adult participants.

But how does SMARTA work and what does it do exactly?
SMARTA is a timer in the shape of a ball, that can be restarted by passing it to another player, and it informs the kid who is currently talking when their turn to talk is about to end, thanks to flashing lights and vibration. The turn duration is decided beforehand by the supervisor of the discussion. When there is more than one player speaking at the same time, the educator can press a dedicated button on the web interface to trigger red light and vibration. At the end of the game, a web page shows statistics about the conversations that took place. The teacher can also consult past statistics, filtered by the team who participated in the discussions.

2 Target groups and User Needs

SMARTA is meant to be used mainly by children (5 to 14 years old) and their educators, but not only. It can indeed be used also by adults in a business setting, like for team building activities or creative brain-storming sessions. However, we mainly focused on the school environment, therefore keeping in mind the needs of children and their educators as well.
The main needs that we identified for this target group are:

- learning to carry out a discussion in a respectful manner, therefore without interrupting other participants
- being engaged in the conversation that is taking place, and encouraged to maintain focus on the activity
- learning to be concise and clear when expressing an opinion, therefore respecting the pre-decided time limitations
- being involved and being given a chance to talk during these group discussions

As mentioned above, SMARTA can be used also by other people, belonging to other contexts and other age groups such as a company's employees during design thinking sessions or team building activities.

In these cases the target groups' needs are slightly different than the previous ones:

- expressing opinions or ideas within a limited amount of time to stimulate creativity and spontaneity
- having one's own idea respected, and being able to express it completely without being interrupted
- incorporating fun, by structuring the discussion as an interactive game in order to stimulate new ideas and constructive team work

Many other groups and scenarios are possible, but our main focus is on children and young teens in a school setting, supervised by an educator.
The goal of our project is to provide the educator a tool to better involve the group of students and receive feedback after the end of the activity, on current and past session. Also, SMARTA aims at providing the students a smart object that helps them handle their turns and feel more engaged in the discussion.

3 State of the art

This section is meant to explore related works and project that address similar problems and issues as SMARTA.

- **CLEAVER – Cluster Estimation and Versatile Extraction of Regions** [1]
  CLEAVER Software has been developed by the company OxfordWaveResearch. It deals with turns in a conversation, but in a more passive and analytical way with respect to SMARTA: instead of regulating a conversation, it is able to distinguish and extract different speakers' speech from an audio file. It also allows to separate actual speech from ambient sound, and to recover overlaps of discussions. This kind of software, however, could be very useful for the future development of SMARTA, if eventually it will include a microphone.

- **Cisco WebEx** [2]
  Cisco WebEx is just one of many different programs available that allow to organize online meetings and webinars. In such contexts, it is sometimes necessary to regulate turns of discussions, and more in particular requests to speak. Through this software, the participants can ask the moderator the possibility to speak by clicking on the "raise your hand" button. This program therefore tries to simulate the practices used in real life, and the control on who is allowed to speak and for how long is completely in the hands of the moderator, who can also decide to mute everyone. The differences with SMARTA are evident: first and foremost the medium is completely different, as SMARTA is a smart

object and Cisco WebEx is instead a software tool; also, SMARTA's approach is to let each speaker decide who will be the next one to talk, therefore allowing the participants a more "distributed" decision process.

- **Pi Ball – The Spherical and Interactive Raspberry Pi 2 Case** [3] This is a project by Youtube user and Electrical Engineer Cabe Atwell. He built a case for the Raspberry Pi 2, in the shape of a ball, and included inside it a LED strip and an accelerometer; the ball would light up whenever it detected motion. Even though this project does not have anything to do with conversations and turn-taking, it is still relevant as it showcases one of the basic working principles of SMARTA: being able to take input from the users in terms of motion of the ball, and showing visual feedback in response. Of course, SMARTA has a more specific function, which is to regulate a conversation, so it also implements the turn timer, and it makes use of vibration as well.

4 Solution – UX Design

4.1 General Approach

While designing the user experience, two things have been kept into clear consideration:

- The system should be simple to use and should require as little complex interaction as possible (button clicks, etc.) from the students;
- To help children feel more comfortable in participating in a class discussion, the smart object should be attractive, engaging and child friendly.

4.2 Details of Interactions and Interfaces

The interactions with the whole system can be divided into two categories, namely interactions involving the teacher and interactions involving the children, which are explained in further details below.

4.2.1 Teacher Interaction

For the interactions concerning the teacher, the system provides a classic interaction based on mouse click inputs and visual effects on a web application. On the web app the teacher can access the configuration page clicking on the "Let's Go" button. They proceed to set the initial parameters. The setup of the smart ball includes definition of the duration of each turn (speaking time), in terms of minutes and seconds, and the choice of the group of students that will be playing, either already in the database or a new one. After the teacher submits these parameters, the children can start the game. The educator can signal when interruptions among players occur by pressing the button “Overlap”, and the smart object will flash red light to warn the players. Once the session is expired, some final information and statistics are available on a final web page and they are also saved in the database. Afterwards, the teacher can consult past statistics, filtered by the team who participated in the discussions.

4.2.2 Children Interaction

The children, unlike the educator, interact only with the smart ball, that detects each throw and manages the conversation through colorful led lights, sounds and vibrations. In order to keep them as focused and engaged as possible, we chose a familiar shape for the smart object: a ball, synonymous with play and fun, able to involve also the shyest children. However, it’s not a common ball because it is able to react during the interaction with the child. We also decided to use standard and quite evocative colors for the light identifying the different game states: green light for the start of a turn, yellow blinking light and vibration for the notification of time running out - this is to signal that the child should pass the ball to someone else, thereby letting them speak - and red light and vibration in the event of an interruption (signaled by the teacher) as it’s commonly associated with mistakes.

4.3 Scenarios

In this section we provide some of the possible scenarios that describe the usage of the system as well as their relative use cases.

4.3.1 Scenario 1: Game Session

There is a class involved in a discussion about an important topic chosen by their teacher: Pollution. They take the SMARTA ball, waiting for the teacher to click on the “Let’s Go” button on the web application. After the educator does this, the ball lights up green. The first student, Martino, takes it in their hands and starts talking, and a timer starts counting down to the time pre-set by the teacher. The child expresses their opinion and as the end of the turn nears, the object starts vibrating and blinking yellow. Martino has to throw SMARTA to a classmate. Once the throw is detected, the new turn starts: the lights become green again and the second child, Tobia, starts talking. Here follows the use case regarding this scenario:

Name	Game Session with all turns respected
Actors	Teacher, class of students
Entry condition	A new game session starts

Flow of events	1. The teacher opens the web app 2. The teacher creates a new session 3. The teacher sets the duration of each turn and selects the group that is playing 4. The teacher presses the Submit button to start the game 5. Martino starts talking holding SMARTA, a new turn starts, green led lights up 6. Martino passes the ball when his turn is about to end, yellow led lights up. The object also vibrates 7. The second child starts to talk holding SMARTA, a new turn starts, green led lights up, the game proceeds
Exit condition	The game session is stopped by the teacher
Exception	Martino doesn't pass the ball before his turn ends: the timer doesn't reset, SMARTA shows steady yellow light and vibration until it is thrown.
Special Requirements	The data collected during the session must be saved and sent to the back-end system

4.3.2 Scenario 2: Game Session 2

A class is involved in a discussion about an important topic chosen by their teacher: Euthanasia. A timer starts for a certain time set up by the teacher. Attilio takes SMARTA in his hands and green leds light up. It's the signal for the beginning of the turn, so Attilio starts talking. He expresses his opinion and as the end of the turn approaches, the object starts vibrating and blinking with yellow lights. Attilio keeps talking and doesn't pass the ball even if his turn is over. To signal this violation of the game rules, SMARTA light up with yellow static lights and vibrates until Attilio decides to throw the ball to the next student. Here follows the use case regarding this scenario:

Name	Game Session: turn not respected
Actors	Teacher, class of students
Entry condition	A new game session starts

Flow of events	1. The teacher opens the web app 2. The teacher creates a new session 3. The teacher sets the duration of each turn and selects the group that is playing 4. The teacher presses the Submit button to start the game 5. Attilio starts to talk holding SMARTA, a new turn starts, green led lights up 6. Attilio keeps talking and doesn't pass the ball even after the yellow blinking lights that signal the approaching end of his turn. 7. After the end of Attilio's turn, SMARTA lights up with steady yellow light and keeps vibrating. 8. Attilio finally throws the ball to the next student. SMARTA lights up green to signal the start of the new turn, and the game proceeds.
Exit condition	The game session is stopped by the teacher
Exception	Attilio passes the ball before his turn is going to end, green led lights up, theimer resets.
Special Requirements	The data collected during the session must be saved and sent to the back-end system

4.3.3 Scenario 3: Game Session 3

A class is involved in a discussion about an important topic chosen by their teacher: "What do I want to become when I grow up?". A timer starts for a certain time set up by the teacher. Checco takes SMARTA, that lights up green, in his hands and starts talking. He expresses his opinion and as the end of the turn nears, the object starts vibrating and blinking yellow. Checco throws it to a classmate. A new turn starts, the object's lights become green again and the second child, Luca, starts talking but then a third child, Ferruccio, talks over them. The teacher recognizes this overlap and proceeds to press the Overlap button. SMARTA lights up with red light and starts vibrating. Here follows the use case regarding this scenario:

Name	Game Session: overlap
Actors	Teacher, class of students
Entry condition	A new game session starts
Flow of events	1. The teacher opens the web app 2. The teacher creates a new session 3. The teacher sets the duration of each turn and selects the group that is playing 4. The teacher presses the Submit button to start the game 5. Checco starts to talk holding SMARTA, a new turn starts, green led lights up 6. Checco passes the ball to Luca when his turn is about to end, yellow led lights up. The object also vibrates 7. Luca takes the ball; a new turn starts. Green led lights up 8. Ferruccio starts talking over Luca 9. The teacher presses the Overlap button. The object lights up with red light and starts vibrating 10. Ferruccio realizes his mistake and stops interrupting, the game proceeds
Exit condition	The game session is stopped by the teacher
Exception	No exceptions
Special Requirements	The data collected during the session must be saved and sent to the back-end system

5 Solution - Implementation

5.1 Hardware Architecture

The project should have been implemented using a Raspberry Pi Zero, which - for its dimensions and low energy consumption - is the perfect solution for this kind of application, built on top of a Linux operating system (Raspbian OS). Unfortunately, due to the COVID-19 Emergency we could not reach the i3Lab to get this component, so we implemented SMARTA with a Raspberry Pi 4 instead. The Raspberry Pi allows to communicate with multiple sensors and actuators through its General purpose input-output (GPIO) connector, which can be easily controlled through high-level system APIs.

5.1.1 Sensors and Actuators

Two sensors and one actuator are connected to the Raspberry Pi, needed to make SMARTA properly working and usable:

- **MPU6050 Accelerometer**: The MPU6050 module consists in a tri-axial linear accelerometer and gyroscope with a full-scale range of $\pm 16g$ with a resolution of 16 bit. For the launch detection algorithm a range of $\pm 4g$ has been chosen as a sufficient range. The module uses the I^2C protocol to communicate with the Raspberry and it is used to detect when the ball has been launched.
- **Vibration motor**: A simple actuator used to interact with the user, as described in section 4.
- **WS2812 LED strip**: LED lights used to notify different messages to the users based on the color of the light.

Another component embedded in the Raspberry Pi and used to interact with the smart object is the internal WiFi chip, that creates an ad-hoc WiFi Access Point by which it's possible to access the web interface of SMARTA. Furthermore, initially we planned to also include a microphone, to detect the occurences of interruptions and overlapping voices. For a more thorough analysis on why we didn't include it in the first prototype of SMARTA, refer to section 5.3.

5.2 Software Architecture

5.2.1 Language

The entire application is written in Python 3, for its easiness to build a working multi-threading application and well organised code structure. Furthermore, having applied an Agile approach to development, Python made the deployment on the target machine very easy and fast for each team member, without the need to compile the source code every time.

While the Web Server is written in Python, the front-end of the web interface has been written in HTML and CSS and interacts with the application through REST APIs.

5.2.2 Frameworks and Tools Used

In order to simplify the development process we chose to adopt some well known and tested frameworks, in particular:

- **RPi.GPIO**, a Python library useful to interact with the GPIO pins;
- **rpi_ws281x**, a Python library for controlling WS281X LEDs;
- **Flask**, a Web Server written in Python used to write REST APIs.

Another tool used to create an ad-hoc Wireless Access Point is **RaspAP**, while the Unix job scheduler **cron** runs the application at startup.

5.2.3 Main Architecture

A finite-state machine approach has been adopted while modelling the design of the application: the machine, `Smarta`, executes its states until an event occurs. Each state is responsible for the event handling during its execution and, if needed, behaves as a transition table returning an appropriate next state to execute.

5.2.4 Launch Detection Algorithm

The launch detection algorithm has been adapted from the fall detection algorithm explained in the conference paper by D. Sprute, A. Pörtner, M. Koenig, and A. Weinitschke, "Smart fall: Accelerometer-based fall detection in a smart home environment" [4], which uses acceleration data measured in g for each axis. The VSA is the vector sum of acceleration, a scalar representing the total acceleration in all three directions and calculated as follows:

$$VSA = \sqrt{x^2 + y^2 + z^2}.$$

In a situation with no acceleration, the axis parallel to the gravitation vector measures $\pm 1g$, while the other axes are close to 0 g. Therefore, the VSA is 1g in such a situation.
The threshold-based launch detection algorithm can be represented by a state chart where each state corresponds to a well-defined state of the launch of the smart ball, as shown in Figure 1. A transition happens if the attached condition is satisfied, while the default and starting state is called "Before launch". For each VSA value, the state machine is updated according to the current state, the input and the transition.

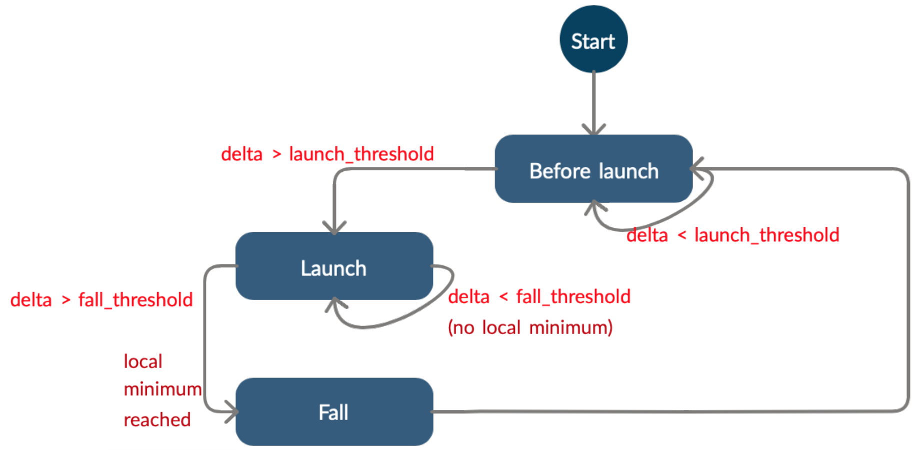


Figure 1: Launch detection algorithm's state chart

The algorithm is described through the pseudo code of the following Algorithms: the first shows how VSA is computed and stored inside a fixed length FIFO stack, the latter describes how the launch detection works, similarly to the state chart of the figure above.

Algorithm 1 How VSA is computed and stored

$vsa\_array \leftarrow []$ ▷ FIFO stack of VSA values
$queue\_length \leftarrow 5$
$acquisition\_period \leftarrow 250\text{ms}$
$accelerometer \leftarrow AccelerometerManager.getInstance()$

while $stopped = False$ **do**
  wait$(acquisition_period)$
  $x \leftarrow accelerometer.getX()$
  $y \leftarrow accelerometer.getY()$
  $z \leftarrow accelerometer.getZ()$
  $vsa \leftarrow \sqrt{x^2 + y^2 + z^2}$
  if $vsa\_array.length \geq queue\_length$ **then**
    $vsa\_array.popFirst()$
  end if
  $vsa\_array.push(vsa)$
end while

Algorithm 2 Launch Detection Procedure

```
launch_threshold ← 0.3
fall_threshold ← 0.15
det_interval ← 200ms
launch_started ← False
launch_detector = LaunchDetector()

procedure Check
    last_vsa ← launch_detector.avg_vsa()
    while last_vsa ≠ None do
        vsa ← launch_detector.avg_vsa()
        if vsa = None then return
        end if
        delta ← |vsa − last_vsa|
        if delta ≥ launch_threshold ∧ ¬launch_started then
            launch_started ← True
        end if
        if delta ≤ fall_threshold ∧ launch_started then
            launch_started ← False                ▷ the ball has been launched
            notify()
        end if
        last_vsa ← vsa
        wait(det_interval)
    end while
end procedure
```

5.3 A note on Audio Analysis for the Detection of Overlapping Voices

In the phase of ideation of our solution, we imagined, among the features of SMARTA, also the possibility to detect overlapping voices. This would have proven useful to evaluate the smoothness and productivity of the conversation, and to give visual feedback in case of overlaps, in order to signal to the participants to stop talking over each other. Also, this feature would have effectively provided a "conversational" aspect to SMARTA, as it would have allowed SMARTA to somehow interact with the conversation. We report the two possible approaches we found to implement this function, and the reasons why in the end we decided not to include this feature at all and make the corresponding visual signals controlled by the teacher through the web interface.

5.3.1 Approach 1 – Decibel Measurement

The first approach involves the measurement of the decibels produced by the voices of the participants in the conversation. It is based on the assumption that if more than one person speak at the same time, the volume of the sound produced will increase noticeably. This implies that the overlapping might be detected simply by comparing the decibels against a given threshold of volume. However, while researching it further, we realized that this was not a viable approach, for the following reasons:

- It is not necessarily true that overlappings correspond to a higher volume of the sound of the conversation, especially since, usually, people stop talking immediately after being interrupted, allowing the conversation to go on at the same volume.
- We don't have prior knowledge of the "base volume" of the participants, and since they are younger kids, they might already speak loudly normally, leading to false positives in the detection of overlappings.
- This system might detect as an overlapping an unrelated raising of voice by the speaker.
- The "base volume" of the conversation might change depending on the context e.g. a quiet classroom, a noisy park, and so on, leading both to false positives and missed detection of actual overlappings.

5.3.2 Approach 2 – Machine Learning

We also considered the possibility to implement a Machine Learning model. It would be trained using data obtained from audio clips of superimposed audiobooks to recreate a situation similar to "more people speaking at the same time", and ideally, given an audio clip as input, it would return a value expressing the number of detected speakers in the conversation. The range of this value would be from 0 to 2, where 0 would mean "no conversation detected", 1 would mean "just one person talking", and 2 "more than one person talking", i.e. two or more, which is our definition of overlapping. Then, this model would continuously analyze the conversation with a very short buffer, and when detecting the value 2, it would trigger SMARTA's previously defined reaction to overlaps. Now, while this model sounds ideal, we found several issues that cause this approach to be unfit for our needs:

- As a team, we don't have any background in Machine Learning and we don't know how to build a suitable data set either, since we don't have any experience with sound analysis and manipulation. This does not mean that we wouldn't be able to learn more about these areas and eventually build this tool, but it would have taken away time and resources from other parts of our project that we felt were much more important: a good user experience being the most valuable to us, both through a usable web interface and through refined and intuitive visual and haptic feedback. However, there are also other more relevant issues with the Machine Learning approach that prove it to be unsuitable to us.

- The level of quality of the voice recording we can obtain is quite low, because of the external material of the ball, which acts as a barrier to the sound, and of the context: the ball is being thrown around, and the participants are not all close to the ball; there might also be background noise. This means that the sound that will be recorded will be muffled and low quality. This could lead to a lower accuracy of the model and skewed results, as the model would be trained using audiobooks, which instead have good voice quality.
- The most relevant issue with this approach is the following: a Machine Learning model is complicated software. We are not even sure if it could work properly on a Raspberry Pi Zero, which is the preferred hardware for our smart object. But even assuming that it would, it would still take several seconds for it to analyze the input and return a value, and these seconds would constitute a critical delay. In fact, most overlappings last less than that. As a result, SMARTA would be able to give visual feedback only several seconds after the fact, possibly resulting in the speakers' confusion about what went wrong. The "real-time" feedback that we would like to implement as we consider it to be the most effective, cannot be achieved through this approach.

5.3.3 Advantages of the teacher-controlled approach

The approach that we eventually adopted doesn't include the detection of the number of simultaneous speakers. Instead, the "overlapping visual feedback" i.e. red light and vibration, is activated by the educator through a button on the web interface. This was done for multiple reasons, the most relevant are the following:

- It simplifies the hardware that is required for SMARTA to work. This allowed us to actually build a prototype of SMARTA, as we didn't have access to a microphone: because of the COVID-19 pandemic, we could not reach the i3lab.
- This approach actually allows to give real-time and accurate feedback to the students, that as we mentioned is essential to make this kind of visual signals effective.
- It allows the teacher to take control of the conversation and play a more active role in it. The role of SMARTA can be extended even to the teacher's own rules. For example, the educator might decide to signal grammar errors, faults in the student's reasoning, filler words or pauses, and so on.

6 Empirical Evaluation

While designing SMARTA, we also planned the empirical research to evaluate the functioning of SMARTA in a real world setting. We have not been able to carry out this research in practice, but we report the theoretical plan of the Empirical Evaluation of our project.

6.1 Research Goal

The main goal of the research is to test in a real world setting whether the usage of SMARTA actually allows for a more balanced conversation as well as understanding if the participants feel more involved in the group activity.

6.1.1 Research Questions

The research questions that we want to answer are the following:

- Can SMARTA improve the distribution of talking time in a group discussion? That is to say, does it allow everyone to speak for a more equal amount of time?
- Can it bring more people to participate in the conversation?
- Can it reduce the number of interruptions?
- Can it make students feel more involved in the discussion?
- Can it increase students' attention to the conversation?
- Does it make the conversation more enjoyable?

6.1.2 Research Variables

The variables that it is necessary to evaluate are here reported, as well as their classification.

- *Number of participants*: quantitative, objective, directly measurable;
- *Number of participants speaking at least once*: quantitative, objective, directly measurable;
- *Length of a participant's turn of speaking*: quantitative, objective, directly measurable;
- *Number of overlaps (interruptions)*: quantitative, objective, directly measurable;
- *Perceived involvement in the discussion*: qualitative, subjective, indirectly measurable;
- *Perceived attention paid*: qualitative, subjective, indirectly measurable;
- *Perceived fun of the discussion*: qualitative, subjective, indirectly measurable.

6.1.3 Metrics

Variables	How the variable is measured
Number of participants	By counting the participants at the start of the discussion
Number of participants speaking at least once	The conversation is recorded, while an AUI student takes notes. Every time a participant speaks, their name and the length of their turn is noted
Length of a participant's turn of speaking	The conversation is recorded, while an AUI student takes notes. Every time a participant speaks, their name and the length of their turn is noted
Number of overlaps	The conversation is recorded, while an AUI student takes notes. Each time an overlapping occurs, it is reported
Perceived involvement in the discussion	A questionnaire is subjected to participants after the conversation. The question is ≪How would you rate your involvement in the conversation?≫. The scale is 1 to 4
Perceived attention paid	A questionnaire is subjected to participants after the conversation. The question is ≪How would you rate how much attention you paid in the conversation?≫. The scale is 1 to 4
Perceived fun of the discussion	A questionnaire is subjected to participants after the conversation. The question is ≪How would you rate the fun you had in the conversation?≫. The scale is 1 to 4

6.2 Participants profiles

Profile	A child of 5-14 years old, who is taking part in a class discussion
Number	Between 5 and 10 for each session

Recruitment	AUI students who take care of this project will ask elementary and middle school teachers if they are willing to let the students participate in this study, by having class discussions with and without the use of SMARTA, with one or more AUI students monitoring the conversations

6.3 Procedure

Type of Study: Controlled study (we examine research variables in the context of class discussions with and without the use of SMARTA, to see whether positive changes occur).

Physical context and spatial settings: A school environment (classroom or gym).

Number of sessions: 10 sessions, 5 with the use of SMARTA and 5 without. Each session is held by a different group of children. The topic of the discussion is always the same (to be decided by the teacher).

Session duration: Each session lasts 20 to 30 minutes.

Protocol: Each session is structured in the following way:

- The teacher briefly introduces the topic of discussion, and either gives SMARTA to the children, or doesn't, if it is one of the control groups.
- The children start discussing the topic, regulating turns autonomously or with the help of SMARTA.
- After about 30 minutes, the teacher stops the discussion.

Data gathering methods:

- A researcher (AUI student) records the conversation and takes notes, and times every turn that occurs, identifying the child's name and the length of their speech. They also report the number of overlaps.
- After the discussion, questionnaires are handed out to the children to rate involvement, attention and fun, or if they're too young, these questions are asked by the teacher and the researchers.
- If the notes of the researcher are not clear enough, later they are compared with the recording of the discussion, to ensure that the data gathered are precise.

Data analysis methods: Once all the data are gathered, the researchers perform a comparison of the average data obtained by the groups that used

SMARTA and the control groups. In particular, the following values are calculated:

- The percentage of children that spoke at least once, over the total.
- The number of overlaps.
- The variance of the length of turns - the smaller this value is, the better the children managed to distribute the conversation, by speaking for similar amounts of time.
- The subjective measurements: involvement, attention and enjoyment.

For each of the calculated values, it is tracked whether SMARTA generated any improvement.

6.4 Other Comments

Some constraints and limitations need to be kept in consideration.

- **Basic assumptions of the study**: in this study it is assumed that all the children are about the same age and that different groups do not have significant differences in terms of the availability of participants to intervene in discussions.
- **Privacy constraints**: it is necessary to ask the children's parents consent for them to participate in this study, as their discussion will be recorded.

7 Value Proposition

This section presents a critical reflection on our work, firstly by analyzing some of the challenges and difficulties we encountered, and then by introducing some other solutions that are also present on the market, and how our product differs from all of them and what is the unique value that it offers.

7.1 Challenges and Critical Aspects

The main challenge we have encountered in developing this project has been to figure out a good solution that could practically improve and complement our natural approach to turn-taking. In the beginning we also considered to implement a software solution for the management of turns in an online video call with many participants, or a conversational interface that recorded and analyzed the conversation and dynamically assigned turns to speakers (for instance, by saying their name). Then, we settled on the idea of a smart object that could be passed around and could also interact with the participants to give them some kind of guidance and feedback on the conversation. The fact that SMARTA could be passed around to indicate who has the right to speak makes the user's interaction with it much smoother and more natural than if they had to somehow indicate the change of turn by, for example, saying their name before talking or through some other kind of "separator" between turns, like a clap. Simply throwing the ball to the next person is much more intuitive. Another critical decision that we had to take is whether we wanted SMARTA to take on a more controlling role in the turn-taking process, like assigning turns to the participants who didn't get a chance to speak yet, or if to make it a more passive tool that didn't actively instruct users on what to do, but helped *them* conduct the conversation better by giving continuous visual and tactile feedback, as well as a final overview of the discussion in terms of number and length of turns. We decided for the latter, as it seemed a more enjoyable and engaging tool to use. Moreover, this solution could be employed also as an educational tool, because it helps users learn how to better use the limited time they have to express their ideas clearly, and to let others speak as well. Lastly, there was a challenge that related to the hardware we used: when we were using the Raspberry Pi 4, we noticed that it tended to overheat after a small amount of time of running SMARTA. This is why we designated as the preferred component the Raspberry Pi Zero.

7.2 Competitors

We researched on the Internet for products similar to SMARTA that are already on the market, and we found nothing exactly like the smart object that we developed. However, we have found many other products that somehow follow the general idea of "smart ball" or "conversational tool" and we report them in the following:

- **Smart Balls to improve sport performance**. One of the most prominent categories of smart balls on the market is the one that consists of smart balls that can track the users' performance in specific sports. They record statistics on metrics like velocity, rotation speed, trajectory and precision. There are examples of this kind of product for various sports, such as soccer (*Adidas miCoach SMART BALL*[5]), basketball (*Wilson X Connected Basketball*[6]) and cricket (*Kookaburra Smartball*[7]). The users can usually access these statistics through a connected mobile app.
- **Other Smart Balls with various purposes**. Other products that we have found still have the shape of a smart ball, but serve a variety of very different purposes. For example, an anti-stress ball that can be used also to interact with games in an associated mobile app (*Xiaomi Yunmai SmartBall*[8]); a ball-shaped robot that can move autonomously and can be used as a safety robot or a fitness assistant (*Ballie by Samsung*[9]); and lastly, an interactive smart ball that is for pets to play with and chase, and can move and vibrate when caught (*GOMI*[10]).
- **Mobile apps to regulate turn-taking**. The last category of products

that we want to present consists of mobile applications that can regulate turn-taking through some very simple interfaces. It is interesting to note that these products are the only ones presented that are aimed at the same target of SMARTA - children and young students. However, there is an obvious difference: these are software tools, while SMARTA is a smart object, with which it is possible to interact physically. Two examples are: *"Turn Taking"* by Switch Access[11], and *"Turn Taker - Social Story & Sharing Tool"* by Touch Autism[12].

7.3 The Value of Our Solution

Our solution, as we have shown, is different from all the other "smart balls" present on the market. The main difference, of course, is the target group - neither athletes, pet owners or technology enthusiasts who want another gadget, but young students and their educators. The second main difference is the need that SMARTA aims to fulfill, namely to help students build a better understanding and proficiency in the skill of constructively participating in discussions.

And why is this a good solution for this purpose? It's quite simple: the way our smart object is used, throwing it like a normal ball, is intuitive and easy to understand also for small children. In fact, the concept of designating an object to give the holder the right to speak already existed[13], and it is in fact quite ancient (it was found for example also in the tribes of indigenous peoples of the Northwest Coast in North America[14]). SMARTA uses this known and effective approach and extends it by warning the students of time running out, therefore helping the students manage their time better, and by providing the teacher with useful statistics at the end of the discussion. Also, we believe that the children's attention is better captured by a ball which can light up, rather than a typical smartphone app or other such tools, as the ball is a physical object that they can hold in their hands and throw around. This also gives the students the impression that they are playing a game, and the "gamification" of the educational discussion activity helps to better engage the students. As a matter of fact, the use of "smart balls" as educational tools has already been explored[15], but it has been noted that while more information is usually beneficial for the educational activity, it might hinder the learner's ability to gauge their own performance, if they become too dependent on the object. This is not the case of SMARTA: our solution takes a more passive role in helping the students, it doesn't actively "direct them" on what to do, and furthermore if the students use it in a correct way, without holding it too long, the ball will not interfere in the communication: it is therefore clear that its final aim is to enable the students to be able to hold a productive discussion also without its visual clues.

8 Future Work

Here are the tasks that we will perform in the future, to improve our prototype:

- **Implement SMARTA with Raspberry Pi Zero.** As we mentioned, we haven't been able to collect the Raspberry Pi Zero from the i3lab because of the COVID-19 pandemic, so the current prototype of SMARTA has been built with the original Raspberry Pi 4. One of the first things we will do is to change this core component into the planned one, the Pi Zero, which will help with avoiding overheating.

- **Add a microphone to collect and analyze conversation data, in order to improve statistics.** In section 5.3 we thoroughly explained our thought process and the reasons that brought SMARTA to not have a microphone. However, we still think that a microphone would be a great added feature and it could prove really useful in collecting data, to add more comprehensive statistics at the end of the conversation.

- **Perform the planned empirical evaluation study**, outlined in section 6.

- **Work on the the external covering.** Finally, we would perform a study of different materials and improve the external covering to optimize three specific characteristics: the robustness of the ball to make it resistant to falls (a likely possibility with the intended target, children), heat dissipation, and last but not least the aesthetic of the ball, to make it more appealing to the young students to play and interact with. A good option to consider is the possibility to 3D print the external shell of SMARTA in such a way to allow the two halves to be screwed together, employing a biopolymer like PLA.

9 Bibliography

References

[1] O. W. Research. Cleaver - oxford wave research, [Online]. Available: https://oxfordwaveresearch.com/products/cleaver/. (accessed: 21.03.2020).

[2] Cisco. Video conferencing - best practices for communicating with participants in cisco webex meetings, [Online]. Available: https://help.webex.com/en-us/nsuwb23/Best-Practices-for-Communicating-with-Participants-in-Cisco-Webex-Meetings. (accessed: 21.03.2020).

[3] C. Atwell. Pi ball – the spherical and interactive raspberry pi 2 case - youtube, [Online]. Available: https://www.youtube.com/watch?v=0ToRKsivivg. (accessed: 21.03.2020).

[4] D. Sprute, A. Pörtner, M. Koenig, and A. Weinitschke, "Smart fall: Accelerometer-based fall detection in a smart home environment", presented at the 13th International Conference On Smart homes and Health Telematics (Geneva, Switzerland), 2015. [Online]. Available: https://www.researchgate.net/publication/276061665_Smart_Fall_Accelerometer-Based_Fall_Detection_in_a_Smart_Home_Environment.

[5] G. Migliorino. Adidas lancia il pallone "smart" che invia i dati all'iphone - iphone italia, [Online]. Available: https://www.iphoneitalia.com/531055/adidas-lancia-il-pallone-smart-che-invia-i-dati-alliphone. (accessed: 04.04.2020).

[6] T. Moynihan. Wilson's new smart basketball magically tracks your stats — wired, [Online]. Available: https://www.wired.com/2015/09/wilsons-new-smart-basketball-magically-tracks-stats/. (accessed: 04.04.2020).

[7] Kookaburra. Kookaburra smartball: Making the ball talk — official kookaburra cricket australia, [Online]. Available: https://www.kookaburra.biz/en-au/cricket/community/news/kookaburra-smartball/. (accessed: 04.04.2020).

[8] L. Armentano. Xiaomi yunmai smart ball: La pallina anti stress di yunmai - gizchina.it, [Online]. Available: https://gizchina.it/2019/08/xiaomi-yunmai-smart-ball-su-youpin/. (accessed: 04.04.2020).

[9] M. Cage. Samsung introduces ballie, the smart ball-shaped robot - somag news, [Online]. Available: https://www.somagnews.com/samsung-introduces-ballie-smart-ball-shaped-robot/. (accessed: 04.04.2020).

[10] Kickstarter. Gomi: The interactive smart ball for your pets by gomilabs — kickstarter, [Online]. Available: https://www.kickstarter.com/projects/gomilabs/gomi-the-interactive-smart-ball-for-your-pets/description. (accessed: 04.04.2020).

[11] A. Inc. Turn taking on the app store, [Online]. Available: https://apps.apple.com/us/app/turn-taking/id879613419. (accessed: 04.04.2020).

[12] ——, Turn taker - social story & sharing tool on the app store, [Online]. Available: https://apps.apple.com/us/app/turn-taker-social-story-sharing-tool/id704876040. (accessed: 04.04.2020).

[13] C. W. G. Jr, "Creative leadership ideas", in. 2011, pp. 38–39.

[14] Wikipedia. Talking stick - wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Talking_stick. (accessed: 11.04.2020).

[15] J. Yi Chow, K. Davids, C. Button, and I. Renshaw, "Nonlinear pedagogy in skill acquisition: An introduction", in. Routledge, 2015, pp. 115–117.