A Creative Game Design and Programming App

Julia Chatain
julia.chatain@inf.ethz.ch
ETH Zurich

Olivier Bitter
bittero@student.ethz.ch
ETH Zurich

Violaine Fayolle
violaine.fayolle@inf.ethz.ch
ETH Zurich

Robert W. Sumner
robert.sumner@inf.ethz.ch
ETH Zurich

Stéphane Magnenat
stephane.magnenat@inf.ethz.ch
ETH Zurich

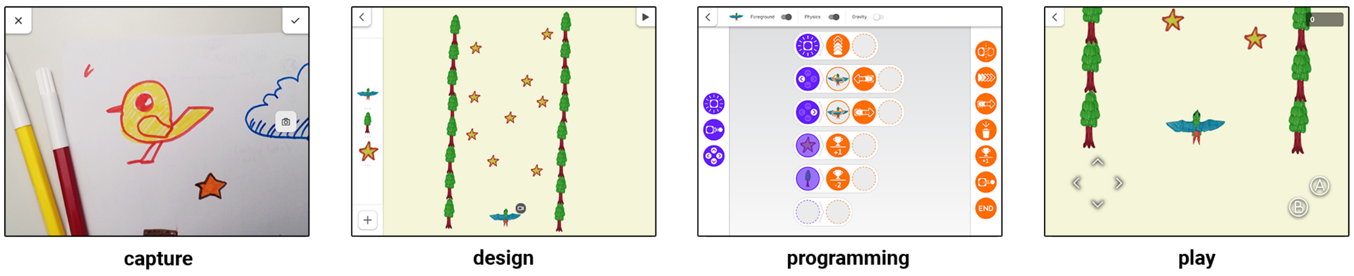


Figure 1: The main phases of game design with our game creation app. Capture: Capture sprites using the device's camera. Design: Assemble a level. Programming: Define the logic of the game in a visual programming language. Play: Playtest the game at any time.

ABSTRACT

We present a game creation app for tablets that builds on the popularity of video games while focusing attention on creativity and problem solving. With our app, users design and build a game by first drawing characters and objects on paper with markers and crayons, and then automatically integrate them with our app. An event-based visual programming language allows to program the game logic. In the spirit of creative play, users can jump at any point between the design, programming and test phases in order to realize their imagination. We evaluate our app with a user study to understand how gender and the use of self-made drawings influence the type of games users create and their state of flow during the process. Our results show that letting users draw their own game elements can lead to higher engagement. We also show that girls tend to spend more time programming and less time testing compared to boys, and that our app can help girls gain self-confidence.

CCS CONCEPTS

• **Human-centered computing → Mixed / augmented reality**; *Empirical studies in HCI*; • **Applied computing → Computer games**; • **Software and its engineering → Visual languages**.

KEYWORDS

creativity, visual programming, content creation, CS education

ACM Reference Format:
Julia Chatain, Olivier Bitter, Violaine Fayolle, Robert W. Sumner, and Stéphane Magnenat. 2019. A Creative Game Design and Programming App. In *Motion, Interaction and Games (MIG '19), October 28–30, 2019, Newcastle upon Tyne, United Kingdom.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3359566.3360056

1 INTRODUCTION

Creative play is both engaging and educational. Finger painting, building with blocks, dancing, or dressing up – all such activities intimately link imagination and self expression with exploration and problem solving. With creative play, children have fun while learning new skills and enhancing their understanding of the world. However, with the popularity and ubiquity of digital devices, an increasing amont of play time happens in the digital world. In this space, video games have proven to be particularly engaging. Although video games easily capture the attention and time of young individuals, their link to creativity is sometimes questionable. Instead of *creative play*, children simply *play*.

We address this situation with an app that builds on the popularity of video games but lies squarely in the space of creative play. Our work is based on the observation that, although *playing* a game may not always be creative, *designing* one certainly is. With our game creation app, users first draw game characters and objects on paper, and take a photo with their tablet to import their drawings into the app as game sprites. Next, they design a game level by instancing objects and characters. Then, they program the behavior and logic of their game using an event-based visual language designed specifically for this purpose. At any point, they can test their game or jump between the design and programming phases in order to realize their imagination. This freeform cycle of design, programming, and play mimics the exploration and problem solving of traditional creative play activities.

In addition to the app design, we also aim at understanding how gender and the use of self-made drawings influence the type of games users create as well as their state of flow during the creation process. Understanding these factors will help shed light on the interplay between creativity, game design, and computational thinking. It will also help us craft better content creation tools and introductory material that make these topics accessible in a gender-neutral way.

An extended version of this paper is available as technical report [Chatain et al. 2019b].

2 RELATED WORK

Previous research has shown that the use of video game mechanics to teach computer science can increase both the learning efficiency and the motivation of learners, for all genders [Papastergiou 2009]. In particular, programming tools that provide visual building blocks, such as Scratch, are effective at teaching key programming concepts to young people, even in the absence of experienced mentors [Maloney et al. 2008]. Recently, as mobile devices have become omnipresent, researchers have identified the value of mobile tools to help spread access to education. [Marcelino et al. 2008].

In this paper, we present our mobile game creation app, following design guidelines for computational thinking tools [Repenning et al. 2010]: we focus on achieving high expressivity in the video game spectrum while keeping a level of complexity low enough to be accessible to beginners and support interaction on mobile devices. Back in the 1980s, [Fenton and Beck 1989] demonstrated a complete system for bringing game creation to end users, through a visual level editor and a textual rule-based programming language. More recently, PlaySketch [Davis and Choo 2014] allows programming game logic from demonstration in a way that bears some resemblance to our visual programming language, albeit with a more limited expressivity. In computer animation, recent studies have explored how to allow beginners to create rich animations, for example by sketching particle systems [Kazi et al. 2014] or expressing virtual forces using brushes [Xing et al. 2016]. These are highly motivating for the user, but do not extend into the realm of scripting.

Related work has demonstrated the potential of combining the physical and the digital worlds. Augmented reality has been used to enhance the creativity of children and to increase their engagement with real-world activities such as coloring [Magnenat et al. 2015b] or music [Zünd et al. 2015]. Using an even more tangible approach, Bloxels [Lee 2016] is a physical bitmap construction kit to design pixel art games. Games are controlled by a combination of predefined game mechanics with limited options for customization. Similarly, when leveraging tangible interactions such as mobile robots and visual languages, children can start developing computational thinking skills early [Magnenat et al. 2015a]. These systems are interesting, but their physical component and their lack of genericity limits their applications. On the contrary, our approach targets common mobile devices as they are available to a large portion of the world's population.

Moreover, creativity is important in the process of learning, and previous studies show that when using tools that let them make their own choices, children are able to learn concepts of higher complexity [Clement 2018] than when choices are imposed. Our app aims at fulfilling this goal as well.

3 THE GAME CREATION APP

3.1 Design

While designing our game creation app, we took inspiration from the guiding design principles of [Resnick and Silverman 2005]. In particular, we considered the following principles:

3.1.1 Lower floor and wide walls. Starting building a game should be easy for children of various ages. It should hence foster acquisition of new concepts and grow in complexity without compromising its low entry barrier.

3.1.2 Simplicity. The app should be as simple as possible without compromising its usability or functionality. It should not require complex knowledge, such as advanced reading skills.

3.1.3 Tinkerability. The app should support incremental development of games and enable children to experiment with new ideas quickly and easily.

3.1.4 Support for self- and classroom-learning. The app should enable self-learning and support classroom activities through intuitive and expected interactions.

3.2 Overview

Figure 1 shows the general workflow of our app. A game is composed of game object *types* and a level, containing *instances* of these types. The *type* contains the sprite and the logic rules of a game object, while the *instance* holds its position, scale, orientation, and velocity. Building a game essentially consists of four phases: capture, design, programming, and play.

3.2.1 Capture phase. To build a game, the user creates new *types* using the device's camera (Figure 1, capture phase). When a white background is detected, an image thresholding algorithm extracts the drawing. Otherwise the image is integrated as is.

3.2.2 Design phase. The created *types* are added to the level editor, onto a palette on the left side of the screen. The editor can be used to assemble the level (Figure 1, design phase). *Instances* can be created by dragging a *type* from the palette to the main area of the level editor. Each *instance* internally links to a *type* and uses its rules, mimicking object-oriented programming concepts. *Instances* can be deleted by dragging them back to the palette. They can be arbitrarily rescaled.

3.2.3 Programming phase. The user can define the behavior of each *type* using a custom event-based visual programming language (Figure 1, programming phase). This language, initially designed for the Thymio robot [Shin et al. 2014], consists of a set of rules, each composed of one event and one or more actions. For each rule, when the event happens, all actions are executed. Table 1 shows all

●	**Start**: When the game starts.
●	**Collision**: When game object collides with another object.
●	**User input**: When the player presses or releases a key.
●	**Mirror**: Flip the game object horizontally or vertically.
●	**Set speed**: Set the speed of the game object.
●	**Add impulse**: Add an impulse to the game object.
●	**Delete object**: Delete the game object.
●	**Update score**: Increase or decrease the score.
●	**Spawn**: Spawn a new object relative to the game object.
END	**End game**: End the game, make the player loose.

Table 1: The event (top) and action (bottom) blocks in our visual programming language.

possible event and action blocks. The parameters of each block are set using a block-specific editor. The game camera can either be attached to an *instance* or set at a fixed position. The zoom factor of the camera and the color of the background can be freely selected.

3.2.4 Play phase. The user can instantly play the game by tapping on the play button on the level editor screen. The play view shows standard game controls: arrows on the left and A/B buttons on the right (Figure 1, play phase). Controls follow the behavior programmed by the user. The ability to instantly play a game at any stage of its development allows fast design-implementation-test cycles, which is critical to create well-functioning gameplay [Ramadan and Widyani 2013].

4 PROGRAMMING LANGUAGE DESIGN

We designed a Visual Programming Language (VPL) inspired by the one used in the Thymio robot [Shin et al. 2014]. To do so, we created 7 target game scenarios of various genres, ranging from platformers to simulations, that our VPL should be able to implement. Each scenario contains a background story, game mechanics, controls, and winning and losing conditions. We identified which *event* and *action* blocks our VPL would require in order to implement these games. We selected a subset of these blocks (Table 1) achieving a trade-off between keeping a simple language, and making it expressive enough to cover a wide span of game mechanics. We also considered the educational interest of the blocks, and the results of different testing sessions where we asked participants to provide feedback regarding their programming process.

5 RESEARCH QUESTION

We aim at understanding whether **our app meets its design goals in terms of accessibility for game creation**, and, if so, **how gender and the creative use of self-made drawings influence the participants' engagement and the created games**.

6 EXPERIMENTAL SETUP

We ran 7 sessions of 80 minutes each, mostly in the area of Zürich, Switzerland, with a total of 104 participants Table 2 (top). In this study, we want to compare the effects of two conditions:

	time	#	age	venue
A	13:00	18	13 – 16	Technorama, Winterthur
B	14:15	23	12 – 14	Kantonsschule ZH Oberland
C	7:30	23	13 – 14	Kantonsschule ZH Oberland
D	11:00	11	8 – 13	Zurich maker day
E	13:00	7	8 – 14	Zurich maker day
F	8:00	12	11 – 13	formatio Privatschule, Triesen
G	10:00	11	12 – 14	formatio Privatschule, Triesen

		condition: drawings by	
time	activity	artist	children
0 – 20 min.	intro., 3 demo. games		
20 – 40 min.	creative session 1	artist	children
40 min.	manikin test 1		
40 – 60 min.	creative session 2	artist	children
60 min.	manikin test 2, duration test		
60 – 80 min.	creative session 3	both artist & children	
80 min.	manikin test 3		

Table 2: Session venues (top) and timing (bottom).

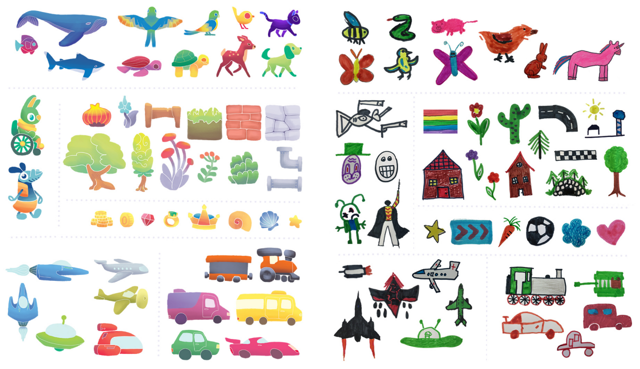


Figure 2: The sprites drawn by a professional artist (left) and an extract of the sprites drawn by the children (right).

(1) sprites drawn by a professional *artist* (Figure 2, left),
(2) sprites drawn by *children* themselves (Figure 2, right).

The participants were split into two rooms, one for each condition, with no communication in-between the rooms. In each room a member of our staff provided help on demand. In addition, another member of our staff was observing the behavior of the participants in both rooms. The observation consisted of noting the current activity of each participant, chosen between: sprite creation, programming, testing, socializing and being distracted.

A session was split into 4 phases of 20 minutes each, as summarized in Table 2 (bottom). During the first phase, the staff member introduced the workshop and demonstrated how to build three simple games. During the next two phases, participants were invited to create their own games freely in creative sessions. During these three phases (60 minutes), participants were strictly assigned to one condition only: either using pre-drawn sprites (artist condition) or drawing their own (children condition). During the last phase,

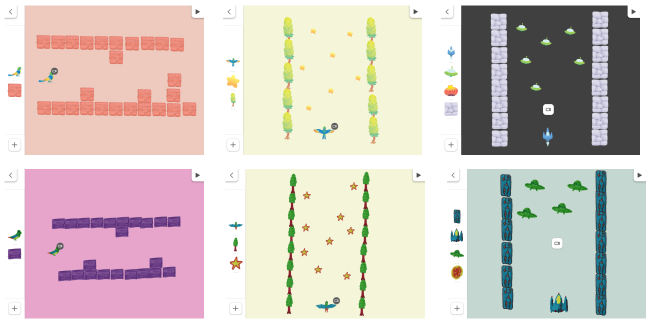


Figure 3: The three demonstrated games; sprites drawn by a professional artist (top) or by a researcher (bottom).

the participants in both conditions were allowed to combine sprites drawn by the artist with sprites drawn by themselves. At the beginning of that phase, our staff member introduced them to the other sprite creation opportunity. After each creative session, an emotional self-assessment test was administered to the participants. In addition, after the second creative session, a duration perception test was also performed.

6.1 Demonstrated games

We demonstrated three games (see Figure 3) of increasing complexity in terms of number of game object types and richness of behavior:

6.1.1 Flappy bird clone. In this action game, a bird moves through a level from left to right with horizontal scrolling. The bird is subject to gravity and must not touch the bottom or top walls.

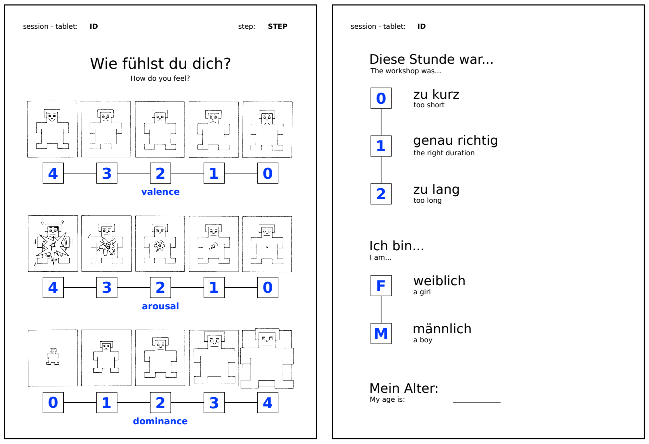
6.1.2 Bird collecting points. In this collection and obstacle avoidance game, a bird moves through a level with vertical scrolling. It must collect stars to gain points and avoid obstacles.

6.1.3 Space invader clone. In this clone of the classic 1978 *Space Invaders* game, the player must shoot aliens and avoid them by moving left and right. The camera is fixed, and the level is composed of a vertical corridor. Waves of aliens slowly zigzag from top to bottom.

6.2 Tests

To measure how participants feel during the activities, we administered two kinds of evaluations. The first one (Figure 4, left) is an existing self-assessment manikin form [Bradley and Lang 1994] that aims at measuring how the participants feel on three axes: 1. valence, how good they feel; 2. arousal, how alert, awake and attentive they feel; 3. dominance, how much in control they feel. For each axis, the participant must tick a box, corresponding to a value from 0 to 4.

In addition, we administered a second test (Figure 4, right) that aims at measuring how quickly participants feel time is passing. Participants must choose one option between "too short", "the right duration", and "too long", that we treat as categorical value. Our goal is to measure whether participants experience a state of flow [Csikszentmihalyi 1997] when using our app. We devised this extremely



session - tablet: **ID** step: **STEP**

Wie fühlst du dich?
How do you feel?

4 — 3 — 2 — 1 — 0
valence

4 — 3 — 2 — 1 — 0
arousal

0 — 1 — 2 — 3 — 4
dominance

session - tablet: **ID**

Diese Stunde war...
The workshop was...

0	zu kurz too short
1	genau richtig the right duration
2	zu lang too long

Ich bin...
I am...

F	weiblich a girl
M	männlich a boy

Mein Alter:
My age is: ______

Figure 4: The forms given to the participants. The emotional self-assessment using manikins (left), and estimation of duration and general information (right). Blue numbers and notes indicate how we measure the results.

age	8	9	10	11	12	13	14	15	16
participant count	2	1	2	12	13	20	44	6	4

Table 3: The distribution of participants' ages.

simple metric after surveying existing flow questionnaires and finding them too complex for the age of our participants and the time they could devote to self-reporting their psychological state. In addition, we asked participants their gender and age, but not their name or any other personal data.

6.3 Game analysis

We collected all 147 created games and measured several features such as the number of games created per user, the number of *types* and *instances* per game, the ratio of scripted ones, and the total number of each block across all object types of a game. In addition, we played all games and classified them in terms of gameplay, presentation, and reward mechanisms. Building on the work of [Lee et al. 2014], we chose the "gameplay" and "presentation" facets, as these are relevant in the simple game engine we provide. We measured the reward mechanisms by checking whether the following behaviors in the game increase or decrease points or trigger a "game over" state: shooting, collision with a character, collision with an item, or passing a finishing line. We chose these because they span the space of dynamics participants have created.

7 RESULTS

Participants were between 8 and 16 years old (Table 3). For all results, we split the data between the *artist* (n = 51) and *children* (n = 53) conditions, and by gender (girl: n = 32, boy: n = 71). For integer values, we used Mann-Whitney U test, and for categorical values, we used Fisher's exact test. In addition to the tests detailed below, we also performed an age analysis, but this one did not provide any significant results. Raw data, excepted the created games due to copyright reasons, are available online [Chatain et al. 2019a].

7.1 Flow

We evaluate flow indirectly by measuring the feeling of participants through self-assessment manikins, their relative perception of time, and their behavior. Table 4 shows that participants feel very happy (3.5 out of 4), moderately awake (2.3 out of 4) and in control (2.6 out of 4). The intensity of all feelings increases or stays constant during the workshop, showing that our app is motivating and engaging for the participants. Participants of the *children* condition are on average more happy and more awake than the ones of the *artist* condition, and these differences also increase during the workshop. We attribute this result to the additional engagement resulting from a more personal contribution in the *children* condition. We see that in average, girls feel less in control than boys. The origin of this gap is difficult to assess. However, this difference decreases during the workshop, which suggests that our app helps girls to gain self-confidence.

All participants found the workshop either too short or the right duration. No participant found it too long. This result, supported by our field observations, shows that the process of creating their own games in our app interests them deeply. The participants of the *children* condition significantly found the workshop shorter. This difference can be due either to a higher engagement or to the reduced time to program their games, as they spent more time creating the sprites as seen in Table 5. Girls spent more time programming while boys spent more time testing (Table 5, right). We observed that, in general, girls tried to have a complete and correct program before testing, while boys tended to use testing to evaluate their mistakes. We believe that this difference could be interesting to explore in a further research study.

7.2 Created games

Most participants (100 out of 104) managed to create non-trivial, functional games. The others failed because they arrived late or had to leave early. Table 6, showing the types of gameplay, presentation, and reward mechanisms, helps to understand this difference. The *children* condition group built more diverse games, including sports games such as a trampoline simulation game, but these games were simpler than the ones of the *artist* condition group. They tended to make more shooting games, but less collection games (Table 6, top). They used a greater variety of game presentation (Table 6, bottom), where the *artist* condition group built horizontal scrollers 80 % of the time. The girls tended to create less variety of gameplay than boys. They seemed to focus on collection games with a variety of objects.

One must be careful when interpreting these results. Three games were demonstrated. In particular, the second one was a collection game, with a very friendly look and world. The third one was a shooting game, in a science-fiction theme and clearly darker and more aggressive. It is therefore likely that we are observing pre-existing societal biases rather than inherent gameplay interest differences. In a further study, one should be careful to build introductory material that minimizes the potential effects of such biases.

entry		artist vs child		p-val.	girl vs boy		p-val.
valence	step 0	3.51	3.62	0.110	3.59	3.57	0.481
	step 1	3.51	3.64	0.098	3.53	3.59	0.197
	step 2	3.52	3.72	**0.049**	3.55	3.65	0.309
	average	3.51	3.66	**0.008**	3.56	3.60	0.217
arousal	step 0	2.02	2.19	0.215	2.19	2.04	0.301
	step 1	2.08	2.47	**0.023**	2.38	2.24	0.472
	step 2	2.30	2.68	**0.033**	2.61	2.45	0.493
	average	2.13	2.45	**0.004**	2.39	2.25	0.328
dominance	step 0	2.47	2.66	0.132	2.34	2.67	**0.020**
	step 1	2.51	2.71	0.162	2.34	2.77	**0.005**
	step 2	2.70	2.85	0.256	2.65	2.87	0.076
	average	2.56	2.74	0.052	2.44	2.77	**0.000**
duration		0.47	0.32	**0.000**	0.47	0.35	**0.000**

Table 4: Self reported feeling of users for different conditions (left) and genders (right). Manikins report valence, arousal, and dominance (higher number means more of the trait). Duration shows expected value, over range 0–2 (0: too short, 1: the right duration, 2: too long). P-value of Mann-Whitney U test (Manikins) and exact Fisher test (duration).

activity	artist vs child		p-val.	girl vs boy		p-val.
asset	5.7 %	27.4 %	**0.000**	16.2 %	17.1 %	0.466
program	59.6 %	44.8 %	**0.000**	57.8 %	49.7 %	**0.010**
testing	18.3 %	13.1 %	**0.009**	12.6 %	17.6 %	**0.032**
distracted	16.4 %	14.7 %	0.386	13.5 %	15.6 %	0.334

Table 5: Percentage of time devoted to different activities for different conditions (left) and genders (right). P-value of Mann-Whitney U test.

gameplay	artist vs child		girl vs boy	
action	73.4 %	45.9 %	80.6 %	51.7 %
shooter	20.3 %	41.0 %	19.4 %	34.8 %
driving/racing	4.7 %	6.6 %	0.0 %	7.9 %
sports	1.6 %	6.6 %	0.0 %	5.6 %
p-value	**0.000**		**0.002**	

gameplay	artist vs child		girl vs boy	
horizontal scrollview	83.1 %	55.7 %	80.6 %	65.6 %
vertical scrollview	10.8 %	23.0 %	16.7 %	16.7 %
static background	6.2 %	21.3 %	2.8 %	17.8 %
p-value	**0.000**		**0.007**	

Table 6: Analyses of game content for different conditions and genders. Type of gameplay (top), game presentation (bottom). P-value of exact Fisher test.

8 DISCUSSION AND FUTURE WORK

The workshops were limited in time. This restriction is potentially an issue as some participants might be slower than others at understanding the system and finding out ideas. Moreover, the *artist* condition requires significantly less time to create the sprites than the *children* condition. We also might have observed a novelty effect, as we only conducted a single workshop. However, the results of this initial study tell a lot about the app ergonomics and how children react to it, and will help guide a further development of the app. We will test the next iterations in more various contexts and environments, with a more diverse panel of users, and using a non-binary evaluation of gender. We would also like to assess the contribution of our app concept to Computer Science education, following previous evaluations [Magnenat et al. 2015a]. Finally, the demonstrated games influenced the creations of the participants, but they were the same in both conditions. Thus, any influence from the game demonstrations should be similar for both conditions and observed differences result from the condition only.

When running the workshops, we noticed that the social component was very important: the participants were sharing their games, asking their friends for feedback, or challenging each other. This social aspect was particularly strong on the *children* condition as they would also share jokes or creative insights about their game world. Moreover, in workshops we organized outside the scope of this study, we noticed that when restricted to one tablet, groups of participants would start collaborating and share the tasks of creating assets, programming, and testing. We believe that it would be interesting to study the influence of social aspects on the involvement and the learning of the users.

9 CONCLUSION

In this paper, we have described a mobile game creation app that allows novices to build their own video game during a 1.5 hour workshop. We have presented an event-based visual programming language adapted to game creation. We have shown that using the app increases users' happiness and engagement, and helps girls gain self-confidence. We have analyzed the differences between the use of users' self-made drawings or sprites made by a professional artist, and shown that the first condition leads to more diverse games, while the second leads to more complex ones. This study proves the value of a mobile game creation app, and provides key findings that shed light on the interplay between creativity, game design and computational thinking. These insights will help crafting content creation tools and introductory material to make these topics accessible to a broad public in a gender-neutral way.

ACKNOWLEDGMENTS

We thank the schools and the Technorama who hosted the experiments, and all participants and their supervisors who took part in this study. We thank Henry Raymond for helping us developing the app and performing the experiments. We thank Dražen Popović for helping us with the result analysis.

REFERENCES

Margaret M Bradley and Peter J Lang. 1994. Measuring emotion: the self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry* 25, 1 (1994), 49–59. https://doi.org/10.1016/0005-7916(94)90063-9

Julia Chatain, Olivier Bitter, Violaine Fayolle, Robert W. Sumner, and Stéphane Magnenat. 2019a. A Creative Game Design and Programming App - raw data. https://www.research-collection.ethz.ch/handle/20.500.11850/364511. https://doi.org/20.500.11850/364511

Julia Chatain, Olivier Bitter, Violaine Fayolle, Robert W. Sumner, and Stéphane Magnenat. 2019b. A Creative Game Design and Programming App - The detailed report. https://doi.org/20.500.11850/365611

Benjamin Clement. 2018. *Adaptive Personalization of Pedagogical Sequences using Machine Learning*. Ph.D. Dissertation. http://www.theses.fr/2018BORD0373/document

Mihaly Csikszentmihalyi. 1997. *Flow and the psychology of discovery and invention.* Harper Perennial, New York.

Richard C. Davis and Kenny T.W. Choo. 2014. PlaySketch: Turning Animation Sketches Into Game Logic. In *IUI 2014 Workshop on Sketch: Pen and Touch Recognition.*

Jay Fenton and Kent Beck. 1989. Playground: An Object-oriented Simulation System with Agent Rules for Children of All Ages. In *Conference Proceedings on Object-oriented Programming Systems, Languages and Applications (OOPSLA '89).* ACM, 123–137. https://doi.org/10.1145/74877.74891

Rubaiat Habib Kazi, Fanny Chevalier, Tovi Grossman, Shengdong Zhao, and George Fitzmaurice. 2014. Draco: Bringing Life to Illustrations with Kinetic Textures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14).* ACM, 351–360. https://doi.org/10.1145/2556288.2556987

Jin Ha Lee, Natascha Karlova, Rachel Ivy Clarke, Katherine Thornton, and Andrew Perti. 2014. Facet analysis of video game genres. *iConference 2014 Proceedings*, 125–139. https://doi.org/10.9776/14057

Kian Teck Lee. 2016. Use of Tangible Learning in Stem Education. In *SIGGRAPH ASIA 2016 Mobile Graphics and Interactive Applications (SA '16).* ACM, 1–2. https://doi.org/10.1145/2999508.3008582

Stéphane Magnenat, Morderchai Ben-Ari, Severin Klinger, and Robert W. Sumner. 2015a. Enhancing Robot Programming with Visual Feedback and Augmented Reality. In *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '15).* ACM, 153–158. https://doi.org/10.1145/2729094.2742585

Stéphane Magnenat, Dat Tien Ngo, Fabio Zünd, Mattia Ryffel, Gioacchino Noris, Gerhard Rothlin, Alessia Marra, Maurizio Nitti, Pascal Fua, Markus Gross, and Robert W. Sumner. 2015b. Live Texturing of Augmented Reality Characters from Colored Drawings. *IEEE Transactions on Visualization and Computer Graphics* 21, 11 (2015), 1201–1210. https://doi.org/10.1109/TVCG.2015.2459871

John H. Maloney, Kylie Peppler, Yasmin Kafai, Mitchel Resnick, and Natalie Rusk. 2008. Programming by Choice: Urban Youth Learning Programming with Scratch. *SIGCSE Bull.* 40, 1 (2008), 367–371. https://doi.org/10.1145/1352322.1352260

Maria Marcelino, Todor Mihaylov, and António Mendes. 2008. H-SICAS, a handheld algorithm animation and simulation tool to support initial programming learning. In *2008 38th Annual Frontiers in Education Conference.* IEEE, T4A–7–T4A–12. https://doi.org/10.1109/FIE.2008.4720530

Marina Papastergiou. 2009. Digital game-based learning in high school computer science education: Impact on educational effectiveness and student motivation. *Computers & education* 52, 1 (2009), 1–12. https://doi.org/10.1016/j.compedu.2008.06.004

Rido Ramadan and Yani Widyani. 2013. Game development life cycle guidelines. In *2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS).* IEEE, 95–100. https://doi.org/10.1109/ICACSIS.2013.6761558

Alexander Repenning, David Webb, and Andri Ioannidou. 2010. Scalable Game Design and the Development of a Checklist for Getting Computational Thinking into Public Schools. In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education.* ACM, 265–269. https://doi.org/10.1145/1734263.1734357

Mitchel Resnick and Brian Silverman. 2005. Some Reflections on Designing Construction Kits for Kids. In *Proceedings of the 2005 Conference on Interaction Design and Children.* 117–122. https://doi.org/10.1145/1109540.1109556

Jiwon Shin, Roland Siegwart, and Stéphane Magnenat. 2014. Visual programming language for Thymio II robot. In *Proceedings of the 2014 Conference on Interaction Design and Children.* https://doi.org/10.3929/ethz-a-010144554

Jun Xing, Rubaiat Habib Kazi, Tovi Grossman, Li-Yi Wei, Jos Stam, and George Fitzmaurice. 2016. Energy-Brushes: Interactive Tools for Illustrating Stylized Elemental Dynamics. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16).* ACM, 755–766. https://doi.org/10.1145/2984511.2984585

Fabio Zünd, Mattia Ryffel, Stéphane Magnenat, Alessia Marra, Maurizio Nitti, Mubbasir Kapadia, Gioacchino Noris, Kenny Mitchell, Markus Gross, and Robert W. Sumner. 2015. Augmented creativity: Bridging the real and virtual worlds to enhance creative play. In *SIGGRAPH Asia 2015 Mobile Graphics and Interactive Applications.* ACM, 21:1–21:7. https://doi.org/10.1145/2818427.2818460