Profiling Styles of Use in Alice

Identifying patterns of use by observing participants in workshops with Alice

Leonel Morales Díaz
Universidad Francisco Marroquín
6ta Calle Final, Zona 10
Guatemala, Guatemala 01010
litomd@ufm.edu

Laura S. Gaytán-Lugo
Universidad de Colima
Km. 9 Carretera Colima - Coquimatlán
Coquimatlán, Colima, México 28300
laura@ucol.mx

Lissette Fleck
Universidad Francisco Marroquín
6ta Calle Final, Zona 10
Guatemala, Guatemala 01010
afleck@ufm.edu

***Abstract*— During workshops on computer animation using Alice, a free platform for three dimensional computer animation created by Carnegie Mellon University, we detected and observed a series of different patterns of use of the platform by attendants. Most participants would start following instructions as precisely as possible. Within a short time some would divert to create visually attractive scenes worrying little about movement and action, others would put two or more characters in the scene and make them talk telling a story, and a fourth group would explore by their own advanced features and functions of Alice well beyond the content of the workshop creating complex and action-rich animations. As these styles kept appearing in every event, a more systematic observation was attempted by designing a form for observers and a survey for participants. Although we have had only one opportunity to test our instruments, we believe that the study is worth continuing and the results may turn revealing not only for Alice but for other block programming environments. In this article we describe the styles observed and the instruments designed for observation. The preliminary results are also discussed.**

Keywords*— *Alice; Computer Animation; Styles of Use; First Programming Environments.

I. Introduction

First programming environments like Scratch [2], Alice [1], Kodu Game Lab [8], Stencyl [9] and several others, are designed to deliver a pleasant experience to kids, youngsters and even adults wishing to learn computer programming. It is important then to analyze the extent to which each one succeeds in achieving that goal.

Equally important is to discover how their users use the tools, what tasks they attempt, the actions they prefer to perform, the outcomes they enjoy obtaining, and the way they relate to others while using the environments. Such discovery can be made on purpose through experiments and controlled settings or by a sort of intuition after noticing patterns of use in different groups and events.

In this paper we will present four profiles of what we call "styles of use" that we detected over time in participants in computer animation workshops with Alice, a free environment for learning to program. After conducting several of such workshops we started noticing that participants engaged with the platform in one of four different styles: instruction follower, scene designer, dialogue storyteller and action animator. In an attempt to validate our observations we designed a survey for participants and a form for an observer to fill for each participant and tested both instruments in one workshop, the results were highly revealing although the number of participants and the setting of the experiment were less than optimal.

We plan to continue with the research in order to find if the identified styles appear consistently across time, diversity of participants, and events, if the descriptions of the styles are clear enough to allow reproducibility of our observations and if there is some tendency in the frequency of observation of each style.

Finally we will argue on the importance of identifying styles of use not only in Alice but in every first programming environment in order to allow the adaptation of teaching and evaluation methods and even make changes in models and designs of these environments.

II. Alice and the Observed Styles of Use

A. Learning to Program with Alice

Alice is an interactive graphical environment for programming three dimensional computer animations that was created and is maintained by Carnegie Mellon University [4]. In the beginning it was thought as a tool for rapid prototyping of 3D animations but soon their creators realized its potential as a first programming environment [3]. It can be freely

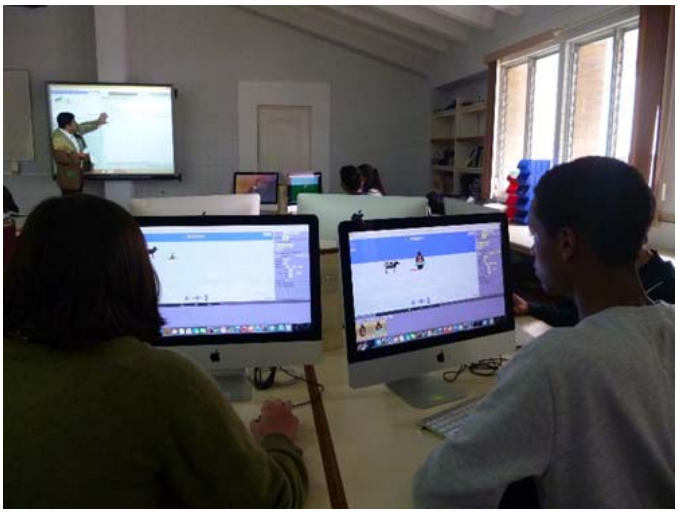

Fig. 1. A workshop on computer animation with Alice.

downloaded and installed in multiple operating systems including Linux, Microsoft Windows and Apple Mac OS. It uses block programming mechanics to produce animations in a 3D space.

In Alice programmers create scenes choosing from predefined stages and adding 3D models from a well populated gallery. Then they script the movements, conversations, property changes and other animation actions by dragging and dropping instructions, methods and functions into the scripting area. Although it is possible to use complex programming structures and algorithms it only requires a few lines dropped to produce interesting and fun animations. As in other block programming environments syntax errors are not possible. All of this makes Alice an attractive and enjoyable platform for learning to program.

However, in announcing our workshops we have avoided the word "programming" and instead used the phrase "Computer Animation with Alice" and the workshops have actually offered instruction on the different forms of computer animations that can be made using Alice. Participants do program, and are introduced to computer programming only within the context of computer animation.

Alice allows learners to test and explore its features in a robust environment of immediate visual feedback. Its friendliness with users and its flexibility to create both simple and complex animations is key for the apparition of different styles of use. In the workshops we often invited participants to create and explore on their own and included free time for that purpose. The combination of a supportive platform and freedom to work on personal ideas can be connected to the "support many paths, many styles" principle, identified by Resnick and Silverman in [10].

B. Instruction Followers

Participants in computer animation workshops with Alice usually start the workshop following the instructions and explanations given. After a while some will divert to exploring the environment on their own, trying and testing the features as they discover them.

Fig. 2. Participant in a workshop adding elements to a scene.

Fig. 3 Time to work freely and a supportive platform enable the apparition of styles of use.

Those that persevere following instructions and paying attention to explanations seem to have been captured by the process and need to continue on it until the end. They are the instruction followers.

Instruction followers try enthusiastically to obtain an animation that is as exact a copy as possible of that the instructor is exposing. They place elements in the scene and in the script in the same position as the instructor, use the same scene settings, and imitate details like the size of the display windows.

Their questions are usually related to how to achieve the same results the instructor obtained: "why is this not working like yours?", "what did you do previously?" or "how did you get that?" They seem to have a sense of accomplishment by producing an animation that looks and behaves like that shown in the workshop. Exploring and discovering by their own is usually delayed until they are sure their work is an acceptable copy and even then they are willing go back to follow instructions when the workshop resumes.

C. Scene Designers

In Alice there is a scene setup mode that lets users add all types of characters and things to the scene making it rich and colorful. It is also possible to use predesigned scenes inspired in popular culture themes and change properties of the ground to make it look like grass, ice, jungle, desert, water, and others.

Some people find this fascinating and start changing and experimenting with different designs of the scene. Alice allows repositioning objects, resize, change color, rotate and other actions, and they seem to have fun doing all that worrying little about scripting actions. In fact they spend most of their time in scene setup mode and go back to edit code mode infrequently.

They may show some interest in learning to script camera movements, a very interesting feature of Alice, because with it they can use camera effects to explore different views of their creation. Alice is for them a sort of artistic composition tool in which they have a digital canvas to work. When they have the

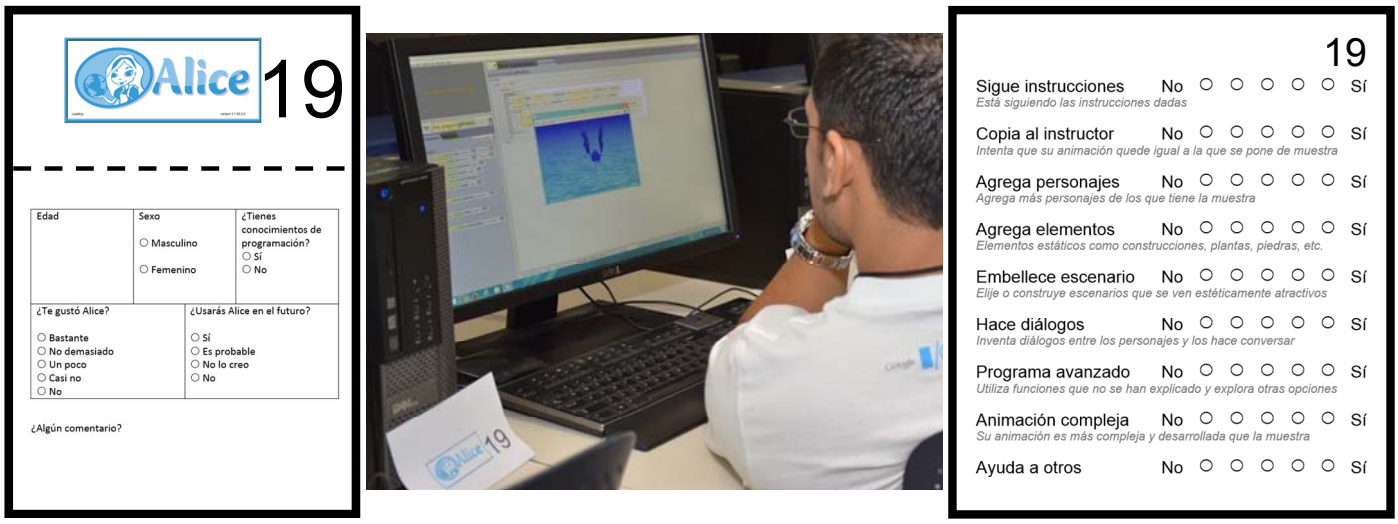

Fig. 4 The survey filled by participants (left) and the observer form (right) both in Spanish. Each participant worked in a station marked with a number (center).

opportunity they ask questions related to how to resize, reposition or change the appearance of objects.

D. Dialogue Storytellers

In Alice it is possible to script objects to say or think something with bubble or cloud callouts that display text messages for a few seconds. The programming instructions to make this are easy to find and some persons start building simple and complex dialogues between two or more characters. They are the dialogue storytellers.

They enjoy, sometimes with visible delight, building these dialogues and watching the resulting animation, adding movement and camera effects only to support the main story. Their code usually contains lengthy sequences of consecutive "say" and "think" instructions.

Working in pairs or in groups goes well with them. They like to receive suggestions for their story and also like letting others watch what they did.

E. Action Animators

In Alice it is possible to employ complex structures of code to produce action-rich animations involving multiple characters and objects. Some discover these features and seem to have a faculty to master them quickly and by their own. They are the action animators.

Action animators enjoy producing elaborated scenes that employ advanced functions and elements not explained in the workshop. For example, certain models in Alice have joints and parts that move independently but need to be coordinated to produce a realistic action as in walk that requires legs steps and arms swings, or wheels that rotate and bump when vehicles move, and so on. Action animators are capable of achieving these effects quickly and create scenes that combine and display them sometimes impressively.

III. Verifiying Observations

As sound and reasonable the detected styles may seem a verification process is needed in order to establish if they can be consistently observed and studied or if otherwise they are an artifact of our workshop methodologies.

In a first attempt to perform the verification we designed a survey to be answered by workshop participants at the end of the event, with basic questions regarding age, sex, previous knowledge of computer programming, the extent to which they liked Alice, how likely they believed it is that they will use Alice in the future and a space for free comments (Fig. 4 left).

In conjunction with the survey a person not enrolled in the workshop would be appointed as observer to collect data in a form with nine qualities to score in a Likert-type scale for every participant: follows instructions, copies instructor, adds characters, adds elements, embellishes scenario, creates dialogues, uses advanced programming, produces a complex animation, and helps others (Fig. 4 right).

Fig. 5. The workshop in which the instruments were tested.

Both tools, the survey and the observation form, were tested for the first time in a workshop in May 2015, and with the collected data we created a model of analysis that produced interesting results. The scores of relevant qualities for each style were used to calculate an index that graded the affinity of the person with the style. After calibrating parameters in formulas and making adjustments we found that classifying a person in a style required an algorithm and not simply selecting the highest index.

With the necessary caution on the preliminary nature of our results, knowing that they come from instruments that need refinement and even other complementary instruments that are not yet designed, we processed the data and present and discuss what we obtained.

A. Preliminary Results

In the following tables we present the results from applying the designed instruments in a workshop on computer animation with Alice with 24 participants. Although all of them answered the survey, 2 left blank the age question and the data for one was discarded because of lack of observation information.

Table I summarizes participants information. Table II shows the raw frequencies observed for each style.

TABLE I. WORKSHOP PARTICIPANTS INFORMATION SUMMARY

Description	Total	Discarded [a]
Answered the survey	24	
Provided information on age	22	
Women	9	
Men	15	1
Had previous knowledge of computer programming	13	1
No previous knowledge of computer programming	11	
Under 20 years old – teenagers	11	
20 years old and above – adults	11	1
Gave Alice highest mark in "how much did you like Alice?" question	15	1
Gave Alice less than highest mark in "how much did you like Alice?" question	9	

[a] Data from one participant had to be discarded because the observation form lacked key information.

TABLE II. RAW FREQUENCIES FOR EACH STYLE

Style	Cases	Percentage
Instruction Follower	11	48%
Scene Designer	3	13%
Dialogue Storyteller	5	22%
Action Animator	4	17%
Total	23	100%

Women exhibited only two of the four styles, instruction follower and dialogue storyteller, while among men the four were observed. Even though the group of women was small, only 9 participants, it accounted for 3 of the 5 dialogue storytellers (60%) as shown in Table III.

TABLE III. FREQUENCIES OF EACH STYLE AMONG WOMEN AND MEN

Style	Women		Men	
	Cases	*Percentage*	*Cases*	*Percentage*
Instruction Follower	6	67%	5	36%
Scene Designer	0	0%	3	21%
Dialogue Storyteller	3	33%	2	14%
Action Animator	0	0%	4	29%
Total	9	100%	14	100%

The previous knowledge of computer programming appears to have little influence in the adoption of a style according to the frequencies observed in both groups and shown in Table IV. The percentage for each style is similar in these two groups.

TABLE IV. FREQUENCIES OF STYLES ACCORDING TO PREVIOUS KNOWLEDGE OF COMPUTER PROGRAMMING

Style	Had Knowledge		No Knowledge	
	Cases	*Percentage*	*Cases*	*Percentage*
Instruction Follower	6	49%	5	46%
Scene Designer	2	17%	1	9%
Dialogue Storyteller	2	17%	3	27%
Action Animator	2	17%	2	18%
Total	12	100%	11	100%

On the other hand, age seems to be important for style selection as can be derived from data in Table V. Younger participants tended much less to adhere to the instruction follower profile while for adults it was the preferred one although the other three were also observed.

TABLE V. FREQUENCIES OF EACH STYLE AMONG TEENAGERS AND ADULTS

Style	Teenagers		Adults	
	Cases	*Percentage*	*Cases*	*Percentage*
Instruction Follower	4	37%	7	70%
Scene Designer	1	9%	1	10%
Dialogue Storyteller	3	27%	1	10%
Action Animator	3	27%	1	10%
Total	11	100%	10	100%

Finally, when grouped according to the mark given to Alice when rating how much they liked it ("how much did you like Alice?" question), we found that the action animator style only

Fig. 6. An observer walked behind participants filling a form with scores in various attributes.

appeared among those that gave the highest grade as shown in Table VI.

TABLE VI. FREQUENCIES OF EACH STYLE ACCORDING TO HOW MUCH PARTICIPANT LIKED ALICE

Style	Liked Alice Most		Liked Alice Less	
	Cases	*Percentage*	*Cases*	*Percentage*
Instruction Follower	6	43%	5	56%
Scene Designer	1	7%	2	22%
Dialogue Storyteller	3	21%	2	22%
Action Animator	4	29%	0	0%
Total	14	100%	9	100%

Again, the preliminary character of these results makes impossible to derive further conclusion although they are important as a baseline and as a comparison set. With new iterations of the application of the instruments after making the appropriate corrections, we hope to identify a tendency and publish findings with better support.

IV. RELEVANCE OF THE PROJECT

First programming environments are powerful tools that support teaching and learning complex computer science concepts. It is thus important to study not only what are the most effective design principles for their construction and evolution but also how their target audiences uses them. Understanding the different styles in which users prefer to engage with these platforms may lead to incorporate features that make it easier to discover the possibilities that match personal interests.

There is a reasonable amount of studies concerning the design, construction and utilization of first programming environments for education or for entertainment, see [5], [6] and [7] for examples. However, the way actual users engage with the environments after initial instruction on their operation has been provided, which features and functions they prefer, or what type of outcomes they attempt is little understood.

When styles of use are fully identified for a platform the instruction methods for prospective users can be adapted to increase efficiency in the education process. It is impossible to tell if the proposed styles for Alice are going to be validated as such but because their identification is the result of observation and attempts of validation that have encouraging results until now, there are good reasons to continue.

There are research questions that could be addressed after the validation is accomplished: do different styles of use require different methods of instruction? Is it possible to identify styles of use for other environments in a similar fashion? Would the styles for other environments be the same as for Alice? And several others.

V. CONCLUSIONS

In this paper we have introduced the description of four styles of use for the Alice graphical environment: instruction follower, scene designer, dialogue storyteller and action animator. The styles were identified after several iterations of computer animation workshops the authors conducted using Alice and patterns of interaction and use were observed in the participants. Grouping the patterns according to similarity led us to profile and describe the styles.

In order to validate that the styles were not an artifact of the methodology used during the workshops and that they can be consistently observed in different settings and with different groups, a pair of instruments were created and tested: a survey for participants that collects basic information and a form to be filled by an observer that registers scores of participants in nine attributes that are later used to classify the person in one of the styles.

Preliminary results obtained from the application of the instruments in one workshop with 24 participants were presented and discussed. The rate of adoption of styles seems to be particularly different between men and women and between teenagers and adults.

There are several limitations that prevent the generalization of the outcomes, but these limitations will hopefully be resolved incorporating corrections and improvements as they are identified and proposed. New instruments may also be created.

We argued on the importance of attending to how users interact with first programming environments on the basis of the foreseeable improvements in instruction methods when they adapt and harmonize with styles of use. As pointed in the article studying users and finding how to they engage with the environment is an area of opportunity for research.

The extension of the search for styles of use to other platforms and programming environments even those that are more advanced and complex was presented as a plausible possibility. The supportiveness of “many paths, many styles” found in Alice can be reasonably expected in several other platforms.

With the iteration of the observation experiment using improved and new instruments in collaboration with other academic institutions we are confident that more reliable data will be obtained and better supported findings will be reported.

REFERENCES

[1] Carnegie Mellon University, "Alice.org," 2015. [Online]. Available: http://www.alice.org/. [Accessed 28 June 2015].

[2] Lifelong Kindergarten Group at the MIT Media Lab, "Scratch - Imagine, Program, Share," 2015. [Online]. Available: https://scratch.mit.edu/. [Accessed 28 June 2015].

[3] M. Conway, S. Audia, T. Burnette, D. Cosgrove and K. Christiansen, "Alice: Lessons Learned from Building a 3D System," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, The Hague, The Netherlands, 2000.

[4] W. Dann and S. Cooper, "Education: Alice 3: concrete to abstract," *Communications of the ACM,* vol. 52, no. 8, pp. 27-29, August 2009.

[5] S. Flanagan, "Introduce Programming in a Fun, Creative Way," *Tech Directions,* vol. 74, no. 6, pp. 18-20, 2015.

[6] M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman and Y. Kafai, "Scratch: Programming for All," *Communications of the ACM,* vol. 52, no. 11, pp. 60-67, November 2009.

[7] S. H. Rodger, J. Hayes, G. Lezin, H. Qin, D. Nelson, R. Tucker, M. López, S. Cooper, W. Dann and D. Slater, "Engaging middle school teachers and students with alice in a diverse set of subjects," in *Proceedings of the 40th ACM Technical Symposium on Computer Science Education*, Chattanooga, 2009.

[8] Microsoft Research, "Kodu Home," [Online]. Available: http://www.kodugamelab.com/. [Accessed 28 June 2015].

[9] Stencyl, LLC, "Stencyl: Make iPhone, iPad, Android & Flash Games without code," [Online]. Available: http://stencyl.com/. [Accessed 28 June 2015].

[10] M. Resnick and B. Silverman, "Some Reflections on Designing Construction Kits for Kids," in *Proceedings of the 2005 Conference on Interaction Design and Children*, Boulder, Colorado, 2005.