E-LEARNING AND ART OF PROGRAMMING: A CONTEXT ORIENTED TO

Guido Averna¹, Biagio Lenzitti¹, Domenico Tegolo¹,²

¹ Department of Mathematics and Computer Science, University of Palermo (ITALY)
² CHAB-Mediterranean Center for Human Health Advanced Biotechnologies, University of Palermo (ITALY)

Abstract

Coding or programming is very important for a number of tasks and this is true not only in problem solving but also in the computer science and over. Many skills have to be acquired before to have a high familiarity degree with this science. In the studies for methods of coding, students have a great problem for understanding on how to solve and to develop algorithms in a rational way, thus the expertise on how to solve and to develop algorithms is the most difficult to acquire for all students in whatever age. This paper introduces the prototype of a framework able to run in the web space and to be supported by different devices and browsers, useful to integrate a number of collaborative contexts for teaching and learning in a workroom, which can be hosted in different academic sites. This framework uses a set of visual objects to define the own program, and to understand how to solve and develop the owner algorithms, and it is in accordance to a blended approach in which students and teacher are cooperating to, it works in a collaborative way and with a distributed approach. Therefore, while a user creates a collection of visual objects, the system develops a text code and its association, so all parts of the diagram will be associated to a text code.

Keywords: Visual framework, Visual Programming Language, VPL, Iconic programming, Coding, Programming.

1 INTRODUCTION

In most programming courses, the standard methods are based on a specific programming language. This approach mix two distinct aspects: learning a programming language and learning to program. The two items are distinct and not the same in some kind of matter. Especially beginners often confuse learning a programming language (in syntax and grammar) with learning to program (the actual, difficult part). Such approach contextualizes learning but a learning out of context is more difficult than learning with relatable context. Initially, students focus just on the syntax, they learn without context, thus they memorize and programming without go together. The art of memorizing kills programming creativity because after having learn the general syntax (which at the starting point is a certain degree of necessary evil), many beginners start learn algorithms in the context of their current programming language and here is exactly where the problem lies. It does not make sense to learn an algorithm in a certain programming language such as the "C", algorithms need to be understood on a conceptual, abstract level independent from programming languages.

One of the main ability to be a skilled programmer is therefore computational thinking. In recent years, the concept of "computational thinking" has inspired new teaching methodologies that consider it one of the main skills to be achieved. According to Wing computational thinking "represents a universally applicable attitude and skill set everyone, not just computer scientists, would be eager to learn and use. Computational thinking is a fundamental skill for everyone, not just for computer scientists. To reading, writing, and arithmetic, we should add computational thinking to every child’s analytical ability” [1].

Computational thinking was the idea that inspired the annual project “Hour of Code” campaign organizes by Code.org® which has engaged 10% of all students in the world, and provides the leading curriculum for K-12 computer science in the largest school districts in the United States.

The language chosen for the development of code.org courses is Scratch [2], a visual programming environments using a more familiar building block command structure [3], and eliminating thorny debugging processes and the risk of syntax errors, too.

Scratch uses a logic of fitting of functions, visual objects, in which experts or students can aggregate several visual objects; any visual objects are fitted to another one. The building block, the flow of data
and the instructions evolution are highlighted only during the run of the task, which is clearer in a flow chart-based environment where the arrows are metaphors of time and flow direction.

A Visual Approach is also used in other development areas such as the Internet of Things (IoT), with development environments such as Node-RED [4] and NETLab Toolkit [5], or for Open Data [6].

This paper introduces the prototype of a framework, supported by different devices and browsers, which is useful to teach and learn in a traditional or collaborative workroom and in different sites (see figure 1a and 1b). Like any other visual programming language, the user will handle the framework in order to create a sequence of visual instructions or objects, and it will be visible to all connected users in real-time. Moreover, it allows a visual program development in a collaborative way by introducing the user leader that, like a teacher, is able to define which other users, like students, have to be involved on development of visual program. Like Google Drive allows to share and to collaborate for drafting a document or creating a spreadsheet, the framework will allow to share and collaborate in drafting a visual program.

This framework will join several proprieties for e-learning in a workroom approach, it works in collaborative and associative way, and it is useful for lessons, exercises and exams, too. The web technologies allow to assign several exercises to students and teacher can see what the students are developing or collaborating.

![Figure 1. a) Traditional mode; b) Collaborative mode.](image)

1.1 Related work

In last years, a number of tools were developed in order to improve the teaching and learning in the field of computer science. Scratch [2] represent one among the most popular free tools to spread the visual programming language. This tool, as many of others, has a visual approach to the programming: the iconic objects are bricks to link, and it is a web tool, too. A person at a time can develop an application, it can use several iconic objects which represent functions or data structures to interact with high level structures (i.e.: with the “go to x .. y..” instruction, an external operator can move to the x and y coordinates). These visual objects can be fitted inside to other blocks and so develop more sophisticated algorithms and methods. Scratch represents an excellent and robust support to develop the computation thinking but it suffers of a real time collaborative develop, therefore an interactive way and a real time visualization of any change among different users it is not allow.

Many other tools share the approach to visual objects with Scratch: AppInventor [7], Blockly [8], are some of them. Blockly is similar to Scratch but it is developed in javascript, and its use allows to manage some kind of devices. AppInventor is a web tool to develop applications for android, it has a mix of visual objects to developing the behavior of an application and it has also a visual representation of object with we must to interact, like a button to push. AppInventor allows developing only for android systems and not for other devices.

These tools allow sharing projects among users, but it is not possible a multiple interaction in real time with the project. Any person learn by self, fitting several visual objects together in altered time.

Another type of visual programming language is LabVIEW [8], it permits developing behaviors of hardware or interacting with the external devices using several high level function. LabView is an application and it does not offer a real time collaboration environment. It allows to store new function,
new programs, but a number of users cannot interact in real time and cannot share with others the changes in a function.

Raptor [9], a Visual Programming Language, is similar to flow-chart. It is a standalone application, and makes available a "flow-chart" approach to develop algorithms. At the best of our knowledge, Raptor is standalone application for Windows, it is a little bit versatile to move icons on the screen thus the application arrange them on the working space in an autonomous way, and it is lacking of real time collaborative approach, too. This meaning that dissimilar devices cannot be used in the same time in different site to collaborate at the developing time.

Each of the previous tools has a different point of view to develop programs and to share and collaborate in the developing phase. In order to do the art of programming own, different points of use and metaphors were introduced; some of them were WEB space oriented and others tuning themselves on specific operation systems.

On the contrary, to previous tools, a tool to teach and learn the art of programming has to put together three different features: WEB modality, real time approach and it need to be supported by the traditional and collaborative way.

Moreover, in order to realize the behavior of algorithms and methods, our framework is a developing system in which users, teachers and students can develop programs through a set of iconic objects: the kernel of our Visual Programming Language.

This overture makes easy to read an algorithm and it abstracts by a specific textual programming language. Finally, in same time it is possible to see text code associate to visual algorithms, today in C language. Moreover, this framework is near to conceptual maps and to UML, and it helps to understand the logic of visual function and to help to familiarize with professional tools using UML, so this tool is also useful to professionals.

2 SIRENE - SHARED INTERACTIVE ENVIRONMENT FOR ENCODING

Frameworks or systems, oriented to improve the learning of "art of programming", design more flexible and reasonable metaphors compared to the more conventional developed software. Software architectures tuned on object-oriented and web-space are able to highlight good potentials to the develop systems on innovative tactics to cultivate good learning on the developing source code.

Although many efforts have been made to develop iconic frameworks [10] in the field of programming languages [11], at the best of our knowledge, the literature shows little efforts on the concept of cooperation. Therefore, the proposed system intends to reduce the gap between the visual language and cooperating programming. This is given to decrease the acquisition time for those concepts own of the programming languages and also to path to final solutions less error-prone.

The proposed framework is a system under developing, based on web application technology and usable for e-learning. It designs a visual tool for teaching and learning the art of programming in workrooms or in different sites, includes a client-server strategy with the support of visual programming language. The significance of visual patterns goes beyond their use and need to extend to their recognition by the user; for example, how to determine what is the correct functionality of each component and its integration with the developing code [12].

The framework allows, in the visual iconic way, growing algorithms and real applications [13]. It can be thinking as a simple and powerful system in which methods can be developed in a collaborative and cooperating style. Its use is preferable to transform methods more schematic and accurate, to analyze each of its steps, and also to represent methods with different metaphors. From the implementation point of view, it gives to the user the possibility to modify the diagram without making any major modifications, to make usable charts in UML. From the usability, it allows having an easy understanding of algorithms, and makes concepts like a map.

In our days, the development of software system is approached like any other products, thus concepts as cooperation and collaboration in teams have to be included during planning phase (phase of development algorithm). At the best of our experience, visual or iconic language approach can give a good chance for a more compressive acquisition of these notions. The framework uses a set of visual objects to define the own functionality, and to understand how to solve and develop the own algorithms. Therefore, this framework allows developing programs using a Visual Programming Language (VPL).
In order to support the highest flexibility possible, structured icons match with textual instruction in a visual fashion. The designed icons are described by a set of formalism, thus each elements can be recombined with other one again.

First of all, a number of features can be ascribed to the system, thus the followings are only a selection: it works in a traditional and collaborative way, a real time approach is supported, a visual interaction way has been included, and it supports a multitasking and distributing approach [13]. “In real time and in interact way” means that any change is immediately visible to other clients and when the framework is in collaboration mode, all students and teacher can modify the project of the algorithm. Moreover, more in deep during the creation of the visual objects for the algorithms, the system develops its code in a textual metaphor and produces its association, so all parts of the diagram will be associated to a text code and the attending students can visualize any change to project.

Another important aspect of the framework regards the elements of the language that the user can customize in order to define the algorithms; given its visual nature, it can be acquired in a fast and naturally way. Moreover, the framework puts in evidence a workspace in which a textual language will appear during the writing of the visual algorithms. It is given for analyzing the same algorithms by other point of view and to produce a textual instance of the visual data model.

Both teachers and students can use this framework, thus a user, normally a teacher, will handle the framework in order to create a sequence of instructions trough visual objects. Moreover, the teacher is able defining which students have to be windowed on the web and able to follow its activities. From the student viewpoint, it can link own job to the web application framework, it can see its and others changes in the web workroom in real time and in interact way. From the teacher landscape, it can define different lesson modality: traditional, collaborative, associative, and it can register the student on each classes. Therefore, other desirable features can be recognized to it: concreteness, immediacy, clarity, visual feedback, easy to use, easy to e-learning, thus real applications can be developed using it.

This framework has been implemented with the Java Script and HTML5 programming language and it generates an Ansi C in the working space, more details of SIRENE framework are reported in Section 3 and 4. Moreover, the framework can generate the visual code in another programming language that will be highlight in the working space.

### 3 HOW SIRENE WORKS

Given the complexity of the proposed framework, and to provide a more suitable representation, a lot of developing system have been investigated and some of them were adopted for a correct design and its appropriate implementation. HTML5, Css, Javascript, Php, Apache, Node.js, Express.js, and Socket.io. Html5, Css and Javascript have been used to define and implement the Visual Programming Language and its data structures.

Any icon can be thinking as a Canvas element of HTML5 with a different shape and color. In addiction no construct icons, "Start" and Linkers have been create for making the flow of a function. Javascript is used for making several functions; they are used to handle local and server functionality and their synchronization. Moreover, Node.js, Express.js and Socket.io improve the performance of the synchronization process. To handle classrooms and to manage a number of projects, Php and Apache are used. They realize a client and server library, written in Javascript, in which communication protocols are used. In addition, several custom libraries are developed to make the correct behavior both client and server. Visual projects will be stored in a special database, thus visual libraries, visual functions and stored working area will context the last visualization of virtual classroom.

The Visual Programming Language is a set of icon. Any Icon is associated to a specific construct of a programming textual language, like "C": Assignment, Loop, Condition, etc. In according to "C" language constructs, icons show different shape, color and action. A "Linker" links some icons to others, and each graphical function uses the "Start" Icon as beginning.

The flow of a visual algorithm follows any Linker from Start icon to the last icon having same nesting level. At this state, only one Start Icon can exist in each function. Icon define a visual function, and its associated code define the real text code. The color of Linkers highlights the nested level in a text code. Therefore, for example the nested Linkers inside a loop have a different color and defines a
nested associated text code. To synchronize to all clients, any change has a specific sending synchronization message. For example, when a teacher or student inserts, removes or modifies an icon than the inserting, a message of removing or modifying will be send to the server. Server checks the modality of lesson and, according to this, sends the equivalent message to all involved clients. When a client generate a new file, the server will send a message to all; this is true for an algorithm, methods and etc, too. The server checks if any message can be eligible to be sent and checks if the client is able send a message before to send it to all.

4 HOW TO USE SIRENE

Experts (teachers) and students use a common web browser to connect to “NOME FRAMEWORK”. A client can make and see the developing of a visual project or visual function. Today, “Nome framework” has two modality of working: traditional and collaborative. In Traditional modality, only teacher can make a project and can make changes and it is able to do available it to attend to the students which can view those changes, too. In some cases teacher can temporarily allow to a student to modify the current project. In collaborative way, both teacher and students can make changes. Some changing can approached both to change to the project and to change in visualization of project. Client side, each change sends a message to the server and receives synchronization messages from it. From server side, it receives changing and sending synchronization messages to other clients, which synchronize the visualization of a lesson. A teacher can realize a traditional or collaborative workroom. Any project can be stored and will can be compiled and downloaded. To define an algorithm you can drag and drop icons you want to use (see figure 2). To create an algorithm you must to create a file using “New File” item menu in main menu. Finally, you can create an algorithm in this new file. The framework use a tab’s structure in which any higher tab identifies the file and any nested tab identifies a function or algorithm. To move to a specific file or function you have to do a click on a tab that identifies the file or function, so you can see icons in that file or function. Any function has two specific other tabs: a tab in which you can insert, modify or delete variables, constant, structure, and go on; and another tab in which you can see the Visual Programming Language and its associated code. Any file tab has other special tabs in which you can see all defined function in that file. In this way, you can open a function clicking its name, and you will see all of the other tabs that could be opened. You can also, save and close projects, files and functions. The teacher can decide if the lesson will be in traditional or cooperative way clicking in modality of lesson, and it can insert and remove attending students to a lesson, too.

Figure 2. The SIRENE Framework: a point of view of the teacher.
5 CONCLUSIONS

The main pioneering contribution of the proposed framework is the capability to develop an algorithm both in a collaborative and simultaneous way. This approach opens up a large number of possible learning scenarios, which can be put in place without changing the learning context. Indeed, the proposed framework will collect several e-learning properties in a workroom approach and in different situations. The collaborative and associative ways will allow fixing the attention of the programming team both in real time and in interactive way. The web technologies allow assigning several exercises to students, and teacher can see what the students are developing or how they are collaborating, thus it can be see useful for lessons, exercises and exams, too. It will be a useful system for both teachers and students in which they can teach and learn the art of programming, they can acquire fundamental skills, to learn how understand how to solve problems, and how to translate their solutions in a programming language. Finally, they can learn how to develop algorithms and programs in an iconic interface. Standard web languages have been used to develop the framework, and this makes easy its integration into Learning Management System like Moodle or Atutor. Finally, in order to verify its usefulness, it will be adopted in two first-level programming courses, both for a degree in Computer Science and in Mathematics. This opportunity will be a great support to delete some kind of mistakes in setting the framework, and other errors given by the developed methodology.

REFERENCES