Scratch Conference 2015

**Workshop** - 60 minutes

**First name:**\*

**Last name:**\*

**Afiliation:\***

**City:** Madrid

**Country:** Spain

**Email:** \*

————

Title: Dr.Scratch

Name of presenter(s): Eva Hu, Mari Luz Aguado

**Summary**

The Dr. Scratch web application is an analytical tool that evaluates your Scratch projects in a variety of computational areas providing feedback. This analyzer is a helpful tool to evaluate your own projects, or those of your Scratch students.

It is suited to students of all ages because results shown are based on the level of Computational Thinking acquired by the students, so that it provides understandable and helpful feedback to the learner.

**Description**

* **Introduction to Dr.Scratch (5min).**

Dr.Scratch is a helpful tool to evaluate Scratch projects. It is powered by Hairball, a static analyzer inspired by Lint that detects frequent programming errors in Scratch projects, such as not initializing the value of a variable or the state of a character, sending messages that are not received by any program, including code that is never run, among others.

* **Typical mistakes or code smells(explaining the variety of computational areas) (10min).**

We’ll explain the typical mistakes or bad programming habits in Scratch that are contrary to the basic programming recommendations:

* Duplicated Scripts: many students repeat scripts in their programs. Thus, abstraction and modularization, two key components of the development of computational thinking, are not trained.
* Sprites naming: students usually don’t change the names of the characters, which are automatically named by the environment  as SpriteX, where X is a incremental number.
* Dead code: tries to find code which has never been executed.
* Sprites attributes: it’s a common mistake change a value of an attribute which hasn’t been initialized.
* **Showing and explaining Dr.Scratch’s results for the three levels (basic, developing, master) (10min).**

Dr.Scratch summaries the output of several plugins in their dashboards showing different information depending of your level of Computational Thinking. That way, Dr.Scratch allows students, independently of their level, to enhance their programming skills.

* Basic dashboard: you can see the aspects of CT with the worst score and some tricks of programming in Scratch.
* Developing dashboard: helps you to understand all the aspects analyzed with Dr.Scratch.
* Master dashboard: displays the same information described in Developing Dashboard including more code smells like Dead Code and Duplicated Scrips.
* **Analyzing your own Scratch projects (15 min).**
* **Signing up for Dr.Scratch (5 min).**
* **Trying out the registered user interface (15min).**