



Lesson plan module 1-2

Objectives:

- Understand key principles of coding.
- Introduction to Scratch
- Develop the concept of computational creation through Scratch programming.
- Develop communication skills, critical thinking, problem solving, collaboration, and creativity skills .

Materials:

- Computers and internet
- Powerpoint Module 1
- Scratch cards (digital or printed)

Duration:

- +/- 2hours

Timing	Teachers activities	Learners activities	Competences
Engage Phase 20' 	• Ask the learners what you can use coding for. Ask for keywords. Write them down in a wordcloud. Place related keywords together. • Suggestions ◦ Use Mentimeter, a Word doc, a big paper or blackboard to make the wordcloud ◦ Show this aftermovie of the exposure day at Zorabots to link coding to the world of work.	• The learners will think about what coding is. • The learners have an understanding about what coding is • Learners think about what they can use coding for (make games, develop websites, calculations, make video`s, apps, robots...)	Critical thinking, brainstorming and communication skills.
Exploration Phase 20' 	Shoe Robot (Unplugged Activity) Unplugged activities help students understand the basics of coding without touching a digital device themselves. In the “Shoe Robot” activity, students practice their skills to construct a code with a well-thought-out sequence. In this activity, as a facilitator, you are a robot and you are 'programmed' by the students, the programmers. The goal of the programmers is to make the robot successfully put on his shoe. Put off your shoe. Then explain to the students that you are a robot and want to put on your shoe. However, you are not programmed to do that. The students are the programmers or engineers and have to make an instruction or code that you can follow.		Critical thinking, collaboration, computational, problem solving skills



Timing

Teachers activities

Divide the learners in 2 groups. Give them a few minutes to discuss and write down what they think is the right instruction to make a robot put on his/her shoe. Then they can tell you the code and you, as a robot, carry out their commands.

Warning: a robot takes everything very literally. When the students instruct you to pick up the shoe, pick it up upside down. When they tell you to put your foot in it, do it in the wrong direction (or to the wrong foot). Make lots of "mistakes" so they understand how accurate programming is.

Example: <https://www.youtube.com/watch?v=2zVpWu1i5gM>

Ask your students what the conclusion is of this exercise.

Learners activities

- The learners discuss and write down the instructions to make a robot put on a shoe.
- When they are ready one learner reads the instruction to the robot.
- the learners understand the key principles of coding
- learners make a conclusion of this exercise:

Ask your students what the **conclusion** is of this exercise. It can be something like this:

- Computers are not clever, but they are very obedient. They will do exactly what you tell them to do.
- Coding is the art of instructing a computer what to do.
- Programming languages are used to communicate to a computer what you want it to do.
- Coding involves feeding the computer step-by-step commands.
- ...

Demonstration

Phase

20'



Demonstration of Scratch

Show your students what Scratch is before you dive into it. You can use the PowerPoint of module 1 and/or share your own screen to showcase some moves in Scratch. You can also show an introductory YouTube video. There are plenty you can use, for example <http://bit.ly/intro-scratch>.

Ask the learners to observe. Maybe you can ask **some questions**:

- What did you see? Does it look difficult?
- (in case you showed a project of yourself) how did it work? What did I do?
- do you see similarities with the shoe robot?

- The learners observe the demonstration and take notes of the following questions :
 - What did you see? Does it look difficult?
 - (in case you showed a project of yourself) how did it work? What did I do?
 - do you see similarities with the shoe robot?

Critical thinking and communication skills



Timing	Teachers activities	Learners activities	Competences
Experimenting Phase 50' 	Now your students are ready to start coding in the most fun way possible: by playing and experimenting. Divide the learners in groups (2-3 learners). Hand out the set of Scratch starter cards, each containing a simple code such as moving with the keyboard arrows, moving with a mouse, changing the color of a character, and so on. Challenge your students to try out as many cards as possible. You can download the cards here: https://resources.scratch.mit.edu/www/cards/en/scratch-cards-all.pdf How to facilitate this experimenting phase: • Walk around and help the students by asking questions to get them to the right answer. • Let groups help each other. Eg: the 'faster' groups can assist the 'slower' ones. • After some time, encourage the groups to experiment (outside the starter cards) • See a great idea? Ask the creator to share with others.	• The learners play around and experiment with Scratch. They can do this individually or in groups • The learners use the Scratch starter cards to explore functionalities.	collaboration, creativity computational, problem solving, coding skills
Reflection Phase 10' 	Ask the learners what they have discovered today and what they would like to do in the next coding club. Guiding questions: • What do you like the best about the project you made? • What did you learn? • What was the hardest part? • What would you like to improve still? • What do you expect from this club?	• The learners share and reflect on the first coding club session. • The learners express their aspirations for the next coding club session.	Reflection, communication skills