

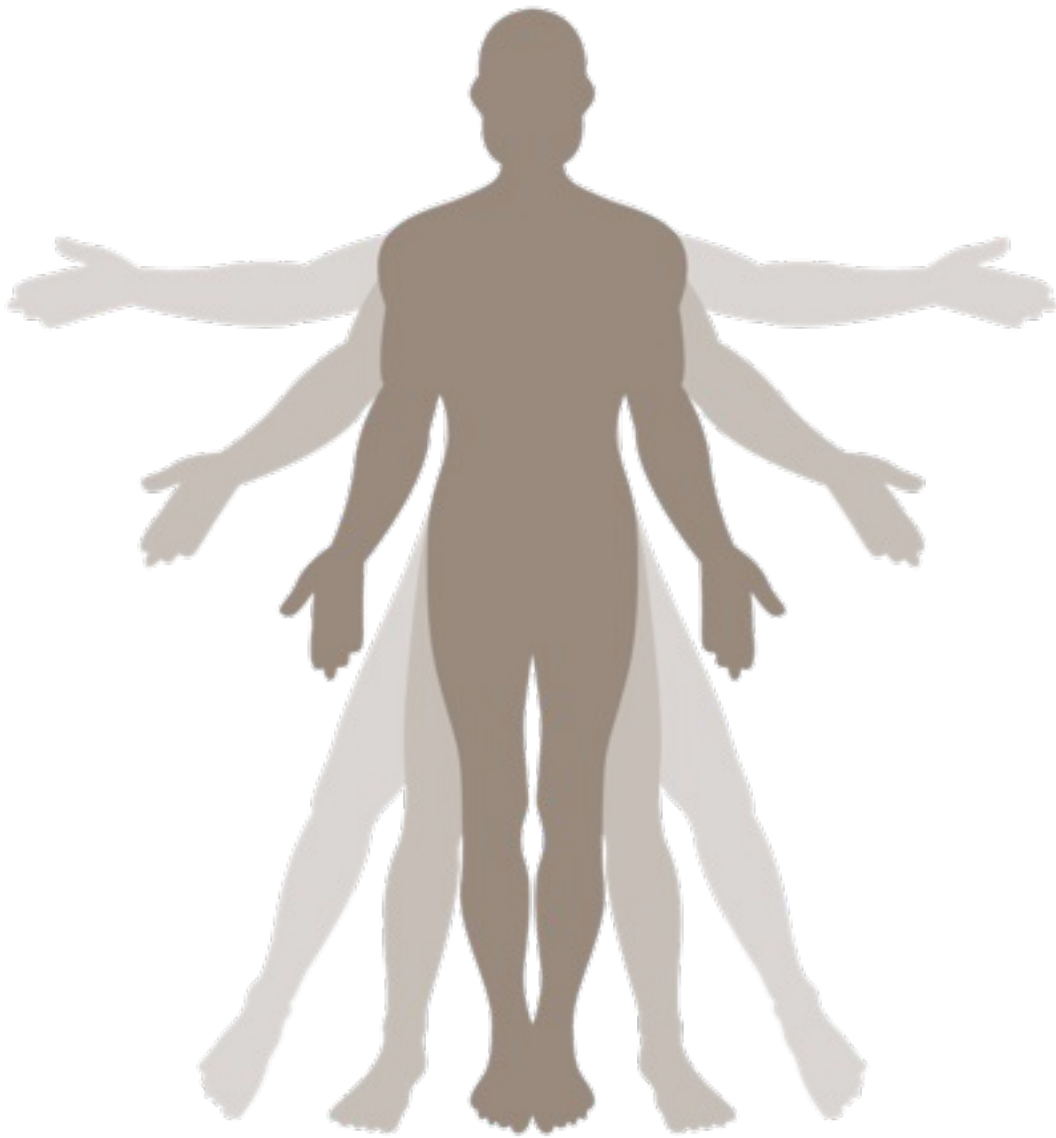
technocamps

Inspiring | Creative | Fun

Ysbrydoledig | Creadigol | Hwyl



Kinect2Scratch Workshop Session Plan



Workshop Schedule

“By failing to prepare, you are preparing to fail.”

Event Set Up	30 minutes
--------------	------------

Introduction

Welcome/Pre-day Forms	5 minutes
Introduction	10 minutes

(Slides 1 & 2)

(Slides 3 & 4)

Kinect2Scratch

Introduction	5 minutes
Scratch Re-cap	10 minutes
Kinect2Scratch	10 minutes

(Slide 5)

(Slide 6)

(Slides 7 - 10)

Pong Game

Sprites	30 minutes
Movement	60 minutes
Ball	60 minutes
Winning	60 minutes

(Slides 11 - 13)

(Slides 14 & 15)

(Slide 16)

Q&A/Post-day Forms	5 minutes
Event Clean Up	30 minutes

(Slides 17 & 18)

Total: 4 hours 15 minutes for attendees

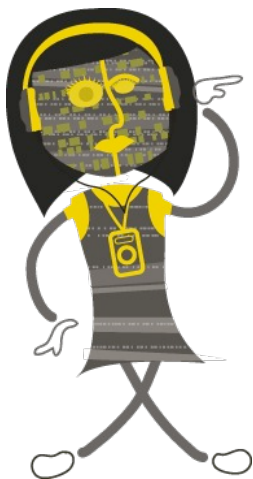
Total: 5 hours 15 minutes for staff

Hardware and Software Requirements

1. Pc/Laptop per participant
2. Projector and powerpoint
3. Scratch software, Microsoft Kinect SDK and Kinect2Scratch
4. Kinect sensor per table/participant



Attendee Prerequisites



1. No programming experience required
2. No OS experience required
3. Completed Consent form - under 16
4. Familiarity with Scratch interface is favourable

Learning Outcomes

1. Understanding and recognising Scratch and Kinect2Scratch.
2. Understand how to utilise programming with input devices.
3. Understand Games Development.
4. Use Scratch to create a simple animated game incorporating image and sound.



Event Set Up and Clean Up

Event Set Up

1. Prepare any pre and post-day questionnaire forms as required. Remember spare pens / pencils.
2. Ensure tables and chairs are arranged to naturally encourage people to sit in groups; ensure no one is sitting with their back to the podium.
3. Test display equipment (e.g. projector) and ensure that presentation and internet connection are working and ready for use.
4. Ensure each laptop has Scratch installed.
5. Ensure each laptop has the Kinect SDK installed, Kinect2Scratch and that the Kinect sensors are successfully connected to the Laptop/PC.



Event Clean Up

1. Ensure all pre-day and post-day questionnaire forms have been collected if required.
2. Safely store away all hardware, charging overnight ensuring all is securely locked away.
3. Clear up litter and refuse. Remember to recycle where facilities exist. Remember to switch off lights, computers, and projectors!



Introduction

Welcome and Pre-day Forms

The first 5 minutes is very much about welcoming and encouraging people to complete any pre-day forms before the workshop begins. Also ensure you read through the pre-day forms with the participants to confirm they have been filled in correctly.

Ensure that you welcome the attendees as they enter the room; this helps to create a positive connection.

Introduction

The introduction gives you time to introduce everyone involved with hosting the workshop.

The main aim is to have everyone settled, focused, and filling required forms, e.g. pre- and post-day questionnaires.

(Slides 1 - 3: Technocamps)

“Good XXX, I’m XXX and I work for an pan-Wales organisation called Technocamps.

Has anybody heard of Technocamps before or been to one of our workshops?

We are a £6 million government funded organisation getting young adults and children aged between 11 and 19 to become excited about Computer Science and what it has to offer, in both their education and future careers.”

Make sure participants complete the pre-day questionnaires before beginning the workshop.

(Slide 4: STEM)

Discuss with the group their views on Science, Technology, Engineering and Maths. What do the pupils think these terms mean? Or what topics do they think are covered within these subjects? Briefly explain to the group what each term means, talking about the link between each of them and how Computer Science is incorporated into each of the other subjects.

(Slides 5 & 6: Scratch re-cap)

Have the participants engaged previously in a Scratch workshop or used this software in ICT lessons in school? What does the group remember about Scratch? Can they remember each

Kinect2Scratch

part of the interface, what it does and how to use it? If participants are struggling to re-call how Scratch works, open the software up (without Kinect2Scratch) and discuss each section of the interface, perhaps depending on available time a basic game can be created using keyboard arrow keys to control a sprite.

(Slides 7 - 10: Kinect2Scratch)

What is a Kinect sensor? Discuss with participants what the sensor is, what it is sensing and what it does with the gathered information. The Kinect2Scratch application allows Scratch to use the Kinect sensor as visual input, a bit like a webcam. To set up the Kinect2Scratch for each Laptop/PC in use, following the instructions below:

- Connect the Kinect via USB to the PC/Laptop.
- Download the Kinect for Xbox “MS Kinect SDK”.
- Run Kinect2Scratch.
- Press “Launch Kinect”.
- Press “Connect To Scratch”.
- Open Scratch, click on “Sensing”, right click on the “Slider Sensor Value” and select “Enable remove sensor connections”.

Kinect2Scratch

When you press “Launch Kinect” the sensor should begin working. On the Kinect2Scratch window 2 video boxes should appear. Adjust the Kinect so you can see your whole body in the videos. You will notice that the left hand image is in shades of red, this means the Kinect has detected you.

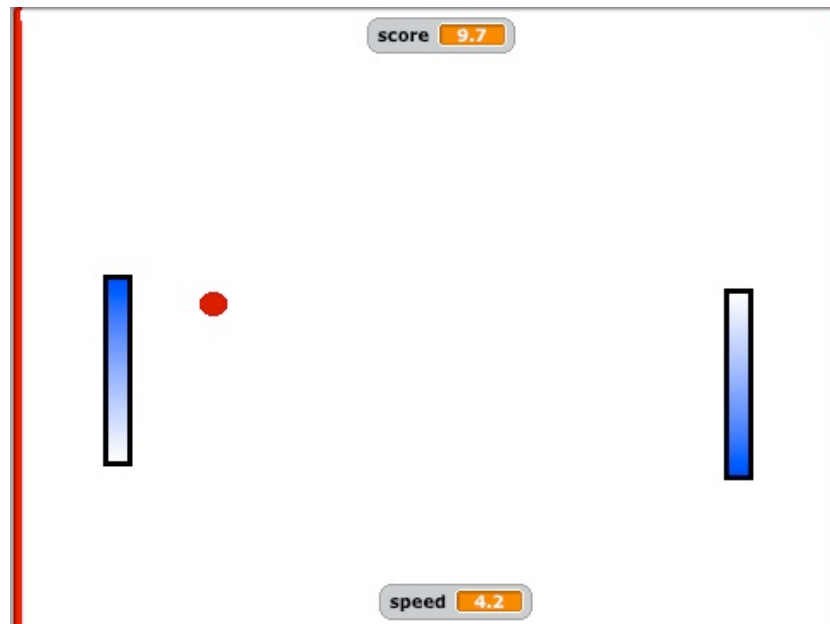
To test some code on the Scratch software implement the following small program:



Kinect2Scratch

(Slides 11 - 16: Paddle Game)

To approach the implementation of a Paddle Game, show the participants an demo of the game they are trying to achieve so they know what to aim for and can compare. The game can be broken down into smaller sections, showing the participants how to achieve a certain feature then allowing them to implement that stage of the game themselves. During the workshop, the groups will be developing a one player tennis game using the Kinect sensor.



There are 2 paddles, the left one will be controlled using the left hand movements up and down, the right paddle will be controlled by the right hand in the same way. A ball needs to be bounced back and forth building up a score. Implement features in the following order:

- **Left or Right paddle:**

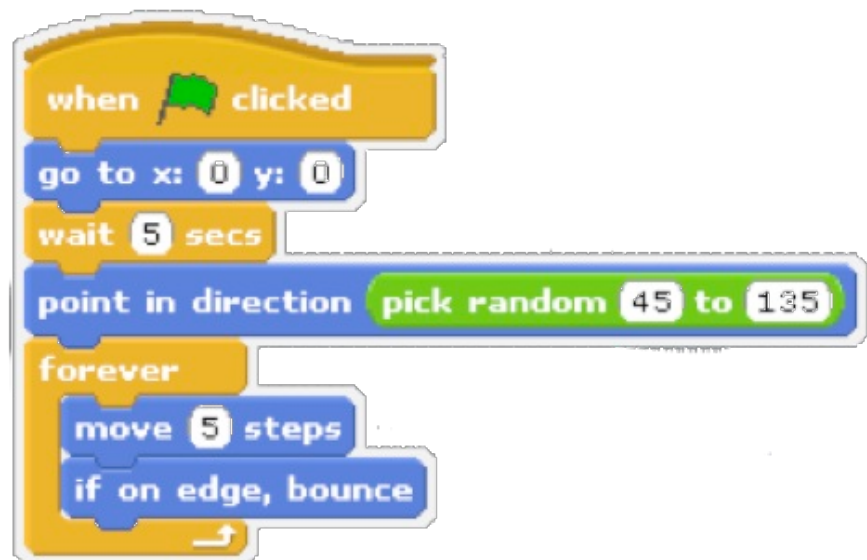
Add a new sprite (you can paint your own paddle or import one). To get the paddle to move, set a default x and y position for “When the green flag is pressed”. Underneath this, add a “forever” loop and inside this add another “go to” command. Depending on where we tell the sprite to go to, it will update the position of the sprite. Under the sensing category, find the “Mouse Y” object and move it to replace the “y” in the “go to” command. This will get the paddle to move at the same height as the mouse pointer without moving left or right on the x axis). Now that we know it works, “mouse y” needs to be replaced with “HandRight Y” or “HandLeft Y” depending on which paddle you are programming.

Don’t forget to implement code for the other Paddle.

Kinect2Scratch

- **The Ball:**

Create the ball sprite using the same method of creating the paddle sprites. Here a wait command will be implemented (adding a pause in your game for a specified amount of time). The code below should be implemented here. It states for the ball to start a co-ordinate 0, 0. It then chooses an angle at random to move in, it also commands the ball how to respond if it collides with an edge of the game.



- **Bouncing the Ball off the Paddles:**

Here you can see how the above code has been expanded further. This implements 2 conditional statements, telling the ball sprite how to respond if it touches the left paddle or the right paddle.



Kinect2Scratch

- **Background:**

The stage is the background for the game, which can be edited by selecting the “Stage” icon and then selecting the “Backgrounds” tab. Here you can either import an image or draw your own. The ball needs to detect when it misses the paddle and hits the edge. To do so, put a red edge on the left hand side of the game and one on the right hand side. When the ball touches the red line, the ball should go back to its starting position in the middle of the game. Implement code for the ball that says the following:

“If the ball touches the red line, then wait for 3 seconds, go to X:0 and Y:0 then point in a random direction)”

This is pseudocode. It is a way of working out code before implementation, planning code using natural language.

- **Score:**

On top of the red lines drawn on the background, draw 2 blank sprites. The “Touching” block under the sensing category can be used to detect which red line is being touched by the ball, where the score is increased depending on the respective line. Make sure you also add a block to reset the score when the green flag is clicked.

- **Increasing Difficulty:**

Suggestions for increasing the game’s difficulty include introducing a curve to the balls movement, having the ball gradually speed up as the game/score progresses or even to decrease the length of the paddles as the game or score progresses.

Kinect2Scratch

(Slides 17 & 18: Finish)

Make sure all participants complete the post-day questionnaire and submit to a Technocamps member of staff. Once this is complete, ensure all work is saved onto machines or onto USB sticks if participants wish to take copies of their work home with them.



Q&A Session & Closure

(Take several questions from the attendees)

(If no questions are asked by attendees, begin summarising topics covered)

Once an appropriate number of questions has been taken, you can then begin to close the workshop. Be sure to have any post-day questionnaire forms filled in by attendees as required. Some audiences may require more prompting to fill in such forms.

Closure and Post-day Forms

The last 5 minutes of closure must be used to, if required, ensure that post-day questionnaires are filled in by the attendees and talk them through the information they have filled in. Ensure that you collect all of the post-day form in before attendees depart, and be sure to thank each person for taking the time to fill in the forms.



technocamps



www.technocamps.com



@Technocamps



Find us on
Facebook

