**Game Design Document - Il mio corpo**

**Game Overview**

Gioco didattico interattivo di Unity distribuito con WebGL. Un programma multimediale progettato per insegnare ai bambini la salute, alimentazione, anatomia e la fisiologia umana. Aiuta i bambini a conoscere il corpo umano con attività interattive del tipo: Di cosa sono fatto, Costruzione del corpo, Smontami.

I bambini impareranno a conoscere ciò che accade nel loro corpo mentre seguono un personaggio scheletrico in 3D appositamente creato.

Caratteristiche aggiuntive:

\* Il personaggio animato dello scheletro 3D funge da “mascotte ”.

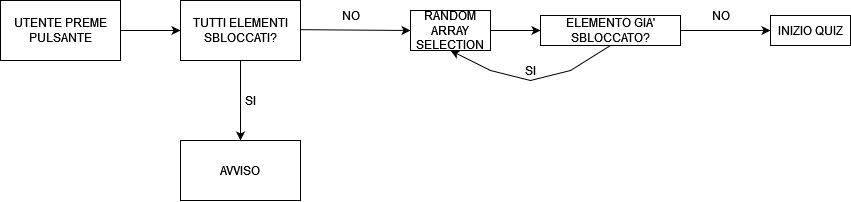
\* Risposte alle domande più comuni sul corpo

\* Divertenti giochi di apprendimento pratico

Di cosa sono fatto?

Consiste nel fare clic su una parte del corpo per saperne di più su di essa.

**Sezione quiz**, domande su parti del corpo scelte in modo random. Logica quiz:



**Audio**

Oltre ad avere una parte testuale per leggere le informazioni su un organo/osso selezionato, il giocatore ha la possibilità di cliccare su un pulsante vocale che darà una lettura audio della casella di testo. Questo serve anche a dare una lettura accurata della pronuncia, aiutando il pubblico più giovane.

Text-to-Audio Source:

www.narakeet.com

Lingua – Italiano

Voce – Gina

How drag drop works.

<https://www.youtube.com/watch?v=pEOetyJ0ULE>

welcome everyone i am savo your game

devil and in today's unity tutorial we

are going to create

a simple drag and drop system that i

will also use to demonstrate

an inventory system and a puzzle game if

that sounds interesting

make sure to stay till the end and see

the whole dragon drop system in action

the first thing you see on my screen is

the setup that i've prepared

i use a canvas with render mode screen

space camera

and i also make sure to have the event

system that comes with it on my scene

it is very much needed in order to

detect our mouse inputs but more about

that in a moment

for the weapons and slots i just use

image game objects and place them

accordingly on the canvas

by the way the swords i'm using are all

created and provided by rappaneum

and if you want to see more of his

amazing work i have a socials and

portfolio in the description

ok back in unity we now create a new c

shop script

named dragon drop or however you like

first of all we are going to need the

library unityengine.event systems

in order to use some interfaces that are

in there

these interfaces are the eye point down

handler

the i begin drag handler and the eye and

drug handler

in case you haven't used interfaces

before you can select them with your

mouse

and you will see the light bulb on the

left go up click on it

and select implement interface and as

easy as that

the whole function just appears and is

ready to use

we are doing the same thing for all

interfaces and to show you how this

works

i will replace the code in the functions

with a simple debug.log

on pointer down we will right click

since this is called

as soon as we click with a mouse on and

drag is called when we release the mouse

click and on begin track is called as

soon as we are dragging the game object

around

ok great so we have set up the basic

system but in order to use it we will

need another interface

called i drag handler which is called on

every frame

we are dragging the sword around

now back in unity don't forget to assign

the script to all the game objects that

you need to use the drag and drop system

for

awesome we are ready for the first test

and i will have the console over here to

see our debug.logs

okay all sorts detect the click and when

i move the mouse the on drag function is

triggered

when releasing it you also see the add

drag

again click on drag and

drag now that our events work correctly

we also need to actually move the swords

with a mouse

with that said jumping back to our

script we used a new react

transform variable and assign it inside

of the start function

next in the on drag function we increase

the anchored position

by the delta of our event data this is

basically the amount of movement we did

with a mouse since the last frame while

holding the game object

now the sword follows our mouse but it

looks kinda strange right

this is because the scale of our canvas

is not set to 1

and that's ok since it needs to adjust

to the screen we are currently playing

on

a quick fix though is to have the canvas

stored in a variable and divide the

eventdata.delta with the canvas.scale

factor

ok now in unity we have to manually

assign the canvas to the public variable

and we are done

alright we fixed the movement pretty

good and our drag and drop system works

great

well to have the swords appear in front

of the slot the game object

should be listed after the slot objects

since the canvas

renders its children based on their

order

ok now it looks as intended and we can

move on to the inventory system

and the puzzle game example for the

first

create a new script i name it slot

script

in here we are using again the

unityengine.event system library

and inherit the eyedrop handler

interface just like

all the other interfaces so far to see

when this is called we can test it with

a debug.log

saying item dropped in unity assign the

new script to all of your slots and hit

play

noticing something yes it's not working

at all

the item drop message never appeared but

why

this happens because as long as we hold

the item with our mouse

it blocks the raycast from hitting

whatever is underneath

so we never actually interact with a

slot

this may looks like a lot of work to fix

but trust me it's pretty straightforward

select all your thoughts and assign a

new canvas group component

there you will see the interactable

boolean which we are going to adjust in

the drag and drop script

in here we get the canvas group

component and on begin drag

we set the block raycast boolean to

false on and drag we reset the block

raycast to true

okay with that done let's see what

happens

great we can click we drag and finally

drop the item in the slot by calling the

write functions

the next step is to properly adjust the

position of the item

inside of the slot so open the slot

script and check if

event data dot drag is not null

point the drag holds the data of the

game object that is currently dragged

and we have

access to whatever component it has

in this case we access direct

transform.anchoredposition and make it

equal to the anchor position of the slot

perfect well almost since the sword

snaps the slot

but the position seems a bit off this is

not the fault of our code though and to

fix it

we open the sprite editor and make sure

that only the sprite we need

is cut out and that the pivot point is

in the middle

now it looks perfect and playing around

with this also is a lot of fun

that's why i also want to show you how

to make a puzzle

game in unity with a drag and drop

system we just created

for the setup i placed a black shadow

version of our sprites

inside of the slots can you already see

which shadow goes to which sword

if yes you must hit the like button

right now and you shall receive a cookie

okay the first thing we need to do is to

hit the like button i mean

assign a new public integer id variable

to both drag and drop script and slot

script

then we check if the id of the sword is

the same as the id of the slot

and if so we will print the message

correct

otherwise we print false let me quickly

give ids from 1 to 4 to all sorts and

the correct id to the slot with the same

sword in it

be careful here since the shadow id

needs to match with the actual sprite id

ok first let's be smart and find the

same form to trigger the correct

printing

now we try again pretending we are dumbs

and there you go

we are false awesome it's time to expand

our unity puzzle game by storing the

starting position

of each sword in a vector 2 variable

with this piece of info we can create a

public function that resets the position

of the game object to its starting

position and you will see why we need

this

in a moment switch to the slot script

and paste the line where we change the

anchored position in the if statement

where both ids are the same

in case they are not we want to snap the

sort back to its original position

and here's where we call the reset

position function from before

you see if we are correct the sword

snaps in the middle of the slot but if

not

it returns to the starting position you

can also

still place them wherever you want in

the scene like in a real puzzle game

until you find out in which slot it

belongs

and that's it i hope you enjoyed the

video and learned how to create your own

inventory or puzzle game in unity

if yes it's really worth it to click on

the subscribe button to see even more

game development videos whenever you

open up youtube

if you have questions i will answer them

in the comments below and to continue

the talk

join our imp unity on discord and become

one of our imps

with that said thank you so much for

watching and i will see you

next time ciao