UD4. Activity 2

CUSTOMIZING THE USER INTERFACE BY USING

CSS STYLES

Toni Barceló García

IES JOAN RAMIS I RAMIS | DAW

Table Content

UD4. Activity 2

Starting from the form made on the activity 1 of this unit, apply **effects** using CSS3:

shadows, **gradients**, **transitions**, **transformations** and animations.

You must apply at least one of each kind of effects.

The final aesthetic design of the form and the use of effects with taste will be

positively valued.

Source Github project

If you clone the project, you will have all the activities I have done so far sorted by folders in "activities" root directory.

Source project:
https://github.com/abarcelogarcia/abarcelogarcia.github.io

Root folder for this activity: UD4. Activity 2
activities/UD4A2_CSS3_Effects

github-pages (latest version)
https://abarcelogarcia.github.io/

Shadow

Input selected & title

I have decided to apply shading to inputs **when they are selected (focus)** and title of the modal header.

```css
input[type=text]:focus,
input[type=password]:focus,
input[type=number]:focus {

    box-shadow: 5px 5px rgba(0, 27, 46, 0.2);
    border-color: var(--gris-corp);
}
```

```css
.modal-header {
    font-family: var(--titulo);
    text-shadow: 2px 0px 0px hsla(0, 0%, 48%, 0.5);
}
```


Gradients

Input selected background

The gradient effect is applied to "inputs" tags **when they are selected (focus)**. The gradient is from left to right with two colours, white and corporate grey that I have defined as a variable in the stylesheet.
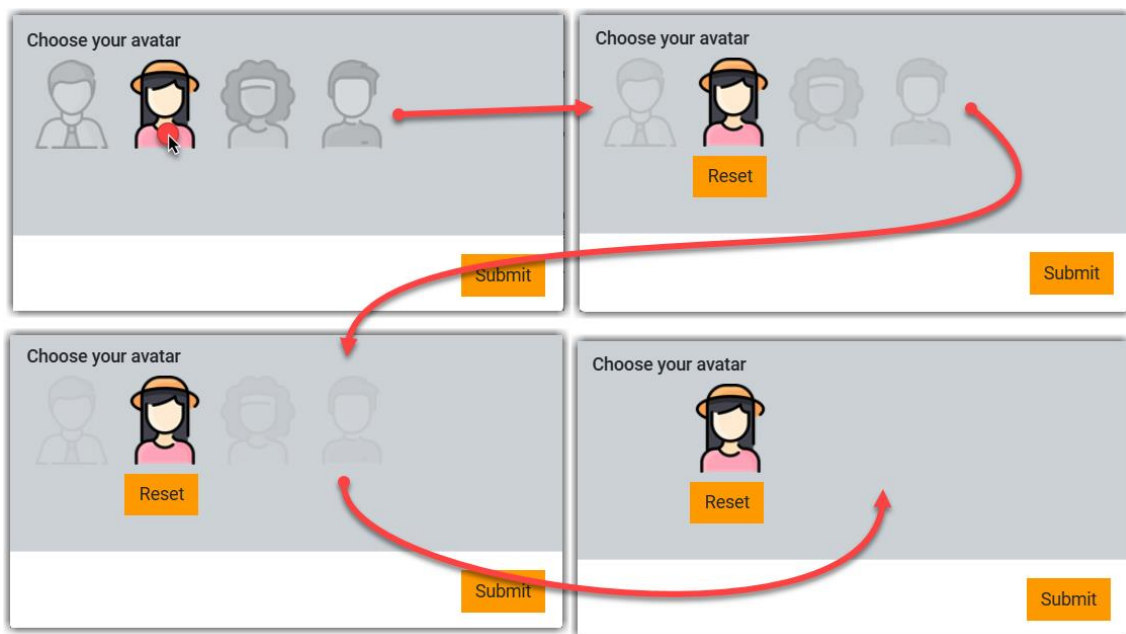
```css
form *>input:focus {

    background-image: linear-gradient(to right, var(--blanco-corp), var(--blanco-corp), var(--blanco-corp), rgba(0, 27, 46, 0.2));

}
```

Transitions

Avatar selected

I have created transitions so that when **one** avatar **is selected**, **the rest** of the avatars disappear. The transition lasts 1 second so that they **disappear smoothly**.
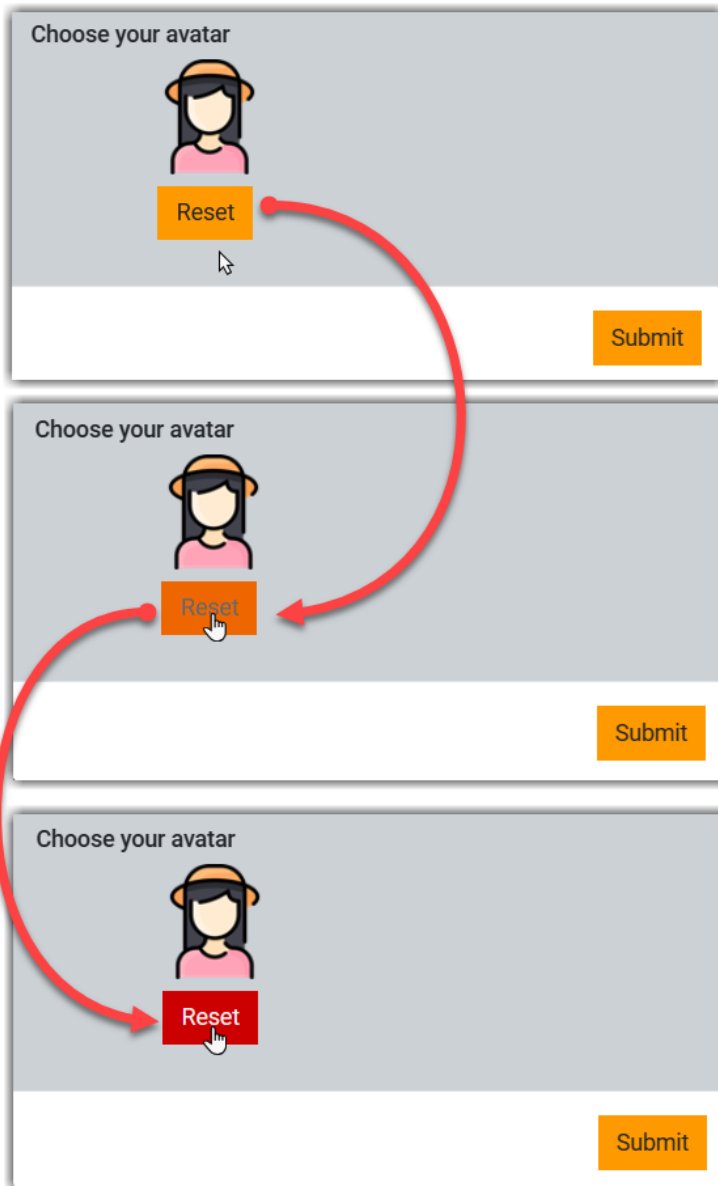


```
/* Hide other avatars */
input[value="2"]:checked~label>:not(#avatar-img2) {
    opacity: 0;
    transition-duration: 1s;
}
```

Reset button on hover

Once you have selected an avatar, a button appears below it to reset the selection. This **button transitions the background color and the text color on mouse-over.**

```
/* Show reset button */
input[value="1"]:checked~div>#reset-btn {
    opacity: 1;

}
```

```
#reset-btn:hover {
    background-color: var(--rojo-corp);
    color: var(--blanco-corp);
    transition-duration: 0.5s;
}
```
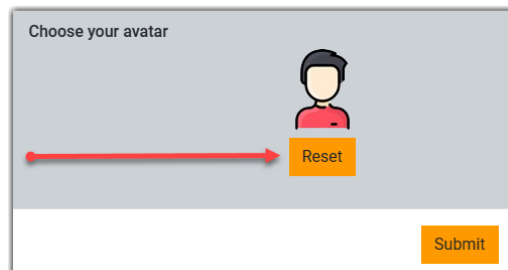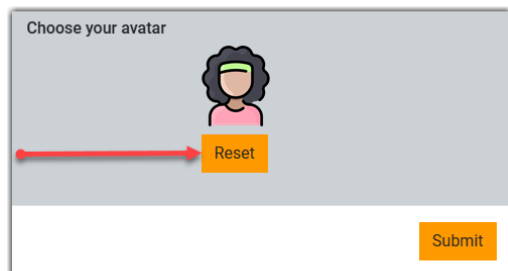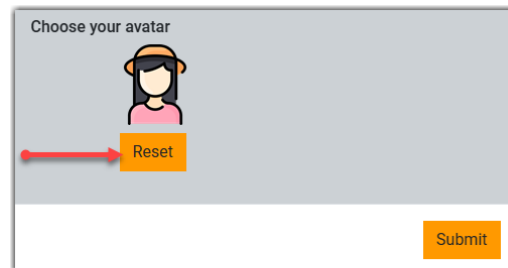
Transformations

Reset button position

When an **avatar is selected**, I must **move (transform/translate) the reset button** so that it appears just below the selected avatar. To do this, I have used transform property with the translate value.

```
/* Show reset button */
input[value="2"]:checked~div>#reset-btn {
    opacity: 1;
    transform: translate(80px);

}
```

```
/* Show reset button */
input[value="3"]:checked~div>#reset-btn {
    opacity: 1;
    transform: translate(165px);

}
```

```
/* Show reset button */
input[value="4"]:checked~div>#reset-btn {
    opacity: 1;
    transform: translate(250px);

}
```
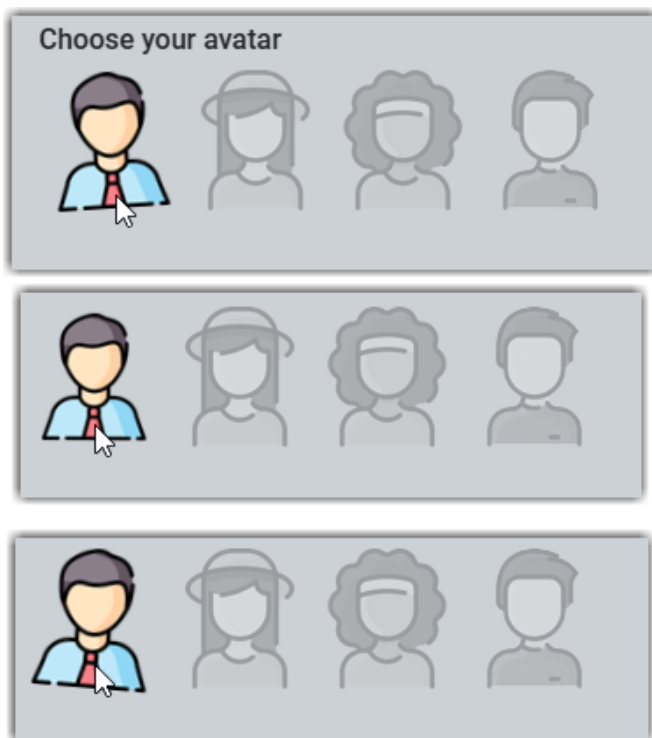
Animations

Vibrating animation

I have made a **subtle animation** that when **hovered over with the cursor**, the **avatar vibrates**. The animation does to appear to vibrate is three steps:

1. Rotates the avatar 5 degrees to the left.
2. Returns to the initial position and reduces its size to 0.9%.
3. Rotates the avatar by 5 degrees to the right.

These steps are executed for 0.1s. and it does it twice.


```css
@keyframes vibrate {
    0% { transform: rotateZ(-5deg); }

    50% { transform: rotateZ(0deg) scale(.9); }

    100% { transform: rotateZ(5deg); }
}
```

```css
/* Delete transparency, grayscale and vibrate animation on mouse hover */
[name*="avatar"]:hover {
    opacity: 1;
    transition-duration: 1s;
    filter: grayscale(0%);
    animation: vibrate 0.1s 2;
}
```