Example 1: Image Slider

(a) Code snippet:

Html:

```
<img src="./img/4.jpg" alt="Me" class="image">
        <img src="./img/4.jpg" alt="Me" class="image">
        <img src="./img/4.jpg" alt="Me" class="image">
        <img src="./img/4.jpg" alt="Me" class="image">
        <img src="./img/5.jpg" alt="Me" class="image">
```

(b) Current result:

This could make it difficult for the buyer to understand the content of the page, where images are displayed sequentially one after the other. There is no method for the consumer to interaction with the images at all or to control the sequence in which they are displayed.

(c) Expected results:

Images that fit the size of the screen, get bigger or smaller as the screen gets smaller, the ability of setting the position of your calculator, as well as the ability to changing the ordering of pictures.

(d) Code snippet:

HTML:

```
<img src="./img/4.jpg" alt="Me" class="image">
    <img src="./img/4.jpg" alt="Me" class="image">
    <img src="./img/4.jpg" alt="Me" class="image">
    <img src="./img/4.jpg" alt="Me" class="image">
    <img src="./img/5.jpg" alt="Me" class="image">
<button onclick="prevImage()">Prev</button>
 <button onclick="nextImage()">Next</button>
```

JavaScript:

```
let slideIndex = 0;
```

```
showImage(slideIndex);


function prevImage() {
  slideIndex--;
  showImage(slideIndex);
}


function nextImage() {
  slideIndex++;
  showImage(slideIndex);
}
function showImage(n) {
  let images = document.getElementsByTagName("img");
```

(e) New results:

The possibility to resize the screen as it is resized, the capability to reorder the images so that the page is under the control of us, and the capability to use the button to alter the background of the whole page.

Example 2: Form Validation

(a) Code snippet:

```
        <label for="fname">First Name:</label>
        <input type="text" id="fname" name="fname"><br><br>
        <label for="lname">Last Name:</label>
        <input type="text" id="lname" name="lname"><br><br>
        <label for="age">Age:</label>
        <input type="number" id="age" name="age"><br><br>
```

(b) Current result:

This can be difficult for the Reader to understand the contents of the page. This is very unspecific and the form parameters are not formalized by system authentication, this means that users can be submitting the shape without giving any or providing inappropriate pieces of information. Essential fields were not valid, which may have resulted in a very ineffective or non-complete commit.

(Desired outcome:

The formation will only be committed if the Form is compliant and if the the Email field contains a valid email format under our manipulation.

(d) Code snippet:

HTML:

```
<form>
      <label for="fname">First Name:</label>
      <input type="text" id="fname" name="fname"><br><br>
      <label for="lname">Last Name:</label>
      <input type="text" id="lname" name="lname"><br><br>
      <label for="age">Age:</label>
      <input type="number" id="age" name="age"><br><br>
      <button type="button" onclick="submitMessage()">Submit Message</button>
    </form>
```

JavaScript:

```
function submitMessage() {
  let fname = document.getElementById("fname").value;
  let lname = document.getElementById("lname").value;
  let age = document.getElementById("age").value;
  if (age < 19) {
    alert("You are not old enough to register for submitting the message");
```

```
  } else {
    alert("Thank you for submitting a message! " + fname + " " + lname + " ");
  }
}
```

(e) New result:

Block the submission of ineffective forms and conduct strict verifications if the compulsory fields would be empty or if a lot we entered is not the valid compulsory required format. The form will only be delivered if the details are all valid and perfectly formatted, and the Name and E-mail fields are non-empty.