Report - Update Feature Toggle Search - 2021-12-08

Accountable: Fernando

**Contextualization**

Responsive design is an important concept on web design. This is the art of making a website look good regardless of the devices.

The adoption of this approach means we adjust website elements, like menu, search box, images, videos, text size, etc., to fit the most important screen sizes used to view the site’s content. The different elements on the screen will require different mechanisms.

We use the resize event listener to detect when the screen dimension is modified. When a specific screen width is reached, we hide some page elements, such as the menu, so we may leverage the canvas in smaller devices more appropriately. For this, we hide some elements, and implement toggles to show and hide them. Currently, it happens when the width is less or equal to 980px. Whenever the viewport’s width is less then 980px, we hide some elements and show their toggles.

That dynamic also requires the way back, i.e, show the elements and hide their togglers when the width is higher than 980px. For this, we still use the resize event handler to detect the when the viewport is more then 980px wider.

The following script takes place to make that dynamic happen

let toggles = Array.from(document.getElementsByClassName('toggle'));

if (window.innerWidth > 980) {

toggles.forEach(function(toggle){

toggle.classList.remove('change');

document.getElementById(toggle.dataset.target).classList.add('show');

})

} else {

// Hide panels and remove toggler active-button flag

let toggles = Array.from(document.getElementsByClassName('toggle'));

toggles.forEach(function(toggle){

toggle.classList.remove('change');

document.getElementById(toggle.dataset.target).classList.remove('show');

})

}

**Problem**

The resize event is fired in unappropriated moments. When the use clicks on the search box on android smartphones, a virtual keyboard is displayed. But as it happens, this feature causes the resize event to fire.

As the resize event fires, and the android smartphone usually offer a viewport that is less or equal to 980px, the instruction is to close all elements and show their toggles. The search box is one of the elements that gets hidden, and thus it disappears when the user tries to type anything on it.

**Solution**

I implemented a mechanism that will not hide the elements when the resize event will not hide the elements while the viewport width is less or equal to 980px. Now, it only happens once when the viewport width gets smaller than 981px for the first time.

Now, the following script takes place:

window.addEventListener("load", function(){

if (window.innerWidth > 980) {

// Detect when it is not over 980 anymore, then add any class

// that keeps the panels visible and remove toggler class

// that indicates it's active.

window.addEventListener("resize", hidePanels, false);

} else {

// Detect when it is not equal or less than 980 anymore,

// then remove any class that keeps the panels visible

// or that indicates that its toggler is active.

window.addEventListener("resize", revealPanels, false);

}

}, false)

function revealPanels() {

if (window.innerWidth > 980) {

console.log("Show Panels")

let toggles = Array.from(document.getElementsByClassName('toggle'));

toggles.forEach(function(toggle){

toggle.classList.remove('change');

document.getElementById(toggle.dataset.target).classList.add('show');

})

window.removeEventListener("resize", revealPanels, false);

console.log('Event removed revealPanels removed.');

window.addEventListener('resize', hidePanels, false);

}

}

function hidePanels() {

if (window.innerWidth <= 980 ) {

// Hide panels and remove toggler active-button flag

let toggles = Array.from(document.getElementsByClassName('toggle'));

toggles.forEach(function(toggle){

toggle.classList.remove('change');

document.getElementById(toggle.dataset.target).classList.remove('show');

})

window.removeEventListener("resize", hidePanels, false);

window.addEventListener('resize', revealPanels, false)

}

}