Referencias <https://www.treinaweb.com.br/blog/efeito-parallax-apenas-com-css/>

<https://willianjusten.com.br/exemplos-de-sites-com-parallax/>

<https://rogertakemiya.com.br/efeito-parallax-com-css/>

<https://developers.google.com/web/updates/2016/12/performant-parallaxing>

<http://css-parallax.webflow.io/>

<https://alligator.io/css/pure-css-parallax/>

https://keithclark.co.uk/articles/pure-css-parallax-websites/

Efeito Parallax

O método de implementação do Parallax é uma técnica que se tornou conhecida a partir do ano de 2013 e muito popular pelo CSS3.

O efeito foi inspirado em games e desenhos clássicos, aonde é possível expressar a profundidade e a perspectiva em um site controlando a quantidade de deslocamento de um elemento específico, principalmente de acordo com o movimento do mouse, acompanhando o usuário enquanto visualiza a página.

Com a ajuda de várias camadas que são sobrepostas umas às outras, é possível criar um efeito tridimensional artificial, podendo aplicar em ícones, imagens e outros elementos.

É possível implementar utilizando o jQuery caso queira alcançar o movimento em velocidades diferentes, porém é mais fácil de demonstrar no CSS3, devido ao uso do javascript/ jQuery carregar lentamente a página, tendo que controlar a posição de cada pixel na tela. Já com o CSS3 é mais simples, deixando o efeito parecido e economiza de problemas de desempenho, permitindo que o navegador ajuste tudo por aceleração de hardware. A taxa de quadros (FPS) permanece estável e a imagem fica suave, podendo ser combinado o parallax com outros chips CSS.

Exemplo

Estrutura

<div class="MainContainer">

<div class="Parallax">

// Conteúdo

</div>

<div class="Content">

<div class="Content1">

// Conteúdo

</div>

</div>

</div>

Estilização

Definimos as propriedades de altura e perspectiva, criando assim um efeito de profundidade

.MainContainer {

perspective: 1px;

transform-style: preserve-3d;

height: 100vh;

overflow-x: hidden;

overflow-y: scroll;

}

/\* Definimos o MainContainer como perspectiva, definindo o cenário para as regras de transformação. \*/

.Parallax {

display: flex;

flex: 1 0 auto;

position: relative;

height: 100vh;

transform: translateZ(-1px) scale(2);

z-index: -1;

}

/\* No Parallax empurramos a div para o segundo plano utilizando o translateZ (-1px).

É possível mudar a distância que deseja empurrá-lo alterando as translateZ regras e as escalas. Isto permite que você mude o quão rápido ou lento seu plano de fundo rolará para fora da vista. \*/

.Content {

display: block;

position: relative;

background-color: white;

z-index: 1;

}

/\* O itens e texto terão metade do tamanho que deveriam ter, colocamos em segundo plano e para compensar adicionamos escala, redimensionando o conteúdo em segundo plano de volta ao tamanho original, dando o efeito de rolagem como se estivesse em segundo plano.  \*/

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Organização e legibilidade do texto OK |  |  |  |  |
| Introdução, motivação e relevância OK |  |  |  |  |
| Descrição e demonstração\* OK |  |  |  |  |
| Analisar aspectos + e - OK |  |  |  |  |
| Conclusão e comentários finais |  |  |  |  |
| Nota\*\* |  |  |  |  |