****

**PROJECT INTRODUCTION**

Github: <https://github.com/yubeizjh/gamattract>

DEMO: <https://gamattract.vercel.app/>

Team member: Junhan ZHU, Xilu WANG, Quji LI

* based on SvelteKit
* Main CSS: w3-css
* Package: svelte-carousl, svelte-forms-lib
* Database: supabase
* Deployed on: Vercel

**CORE coding**

1. Document manage

**图形用户界面, 文本, 应用程序

描述已自动生成** Classical Svelte folder, clear and simple

1. **Layout page – Navigator for small screen**

<!-- Navbar on small screens (remove the onclick attribute if you want the navbar to always show on top of the content when clicking on the links) -->

<div id="navDemo" class="w3-bar-block w3-black w3-hide w3-hide-large w3-hide-medium w3-top" style="margin-top:46px">

<a href="/" class="w3-bar-item w3-button w3-padding-large" on:click={myFunction}>LATEST</a>

<a href="/gallery" class="w3-bar-item w3-button w3-padding-large" on:click={myFunction}>COLLECTION</a>

<a href="/gameFeeds" class="w3-bar-item w3-button w3-padding-large" on:click={myFunction}>FEEDS</a>

<a href="/moments" class="w3-bar-item w3-button w3-padding-large" on:click={myFunction}>BEST MOMENTS</a>

<a href="/websiteGuide" class="w3-bar-item w3-button w3-padding-large" on:click={myFunction}>WEBSITES</a>

<a href="/about" class="w3-bar-item w3-button w3-padding-large" on:click={myFunction}>CONTACT</a>

</div>

<!-- Small Screen Navigator -->

<script>

function myFunction() {

var x = document.getElementById("navDemo");

if (x.className.indexOf("w3-show") == -1) {

x.className += " w3-show";

} else {

x.className = x.className.replace(" w3-show", "");

}

}

</script>

More button

<div class="w3-dropdown-hover w3-hide-small">

<button class="w3-padding-large w3-button" title="More">MORE <i class="fa fa-caret-down"></i></button>

<div class="w3-dropdown-content w3-bar-block w3-card-4">

<a href="https://gameattract.wordpress.com/" class="w3-bar-item w3-button" target="\_blank">GAMEATTRACT FEEDS</a>

</div>

</div>

1. **Index Page – using package “svelte-carousl”**

<svelte:component

this={Carousel}

bind:this={carousel}

autoplay

autoplayDuration={3000}

autoplayProgressVisible

>

..

</svelte:component>

<script>

import { onMount } from 'svelte';

let Carousel; // for saving Carousel component class

let carousel; // for calling methods of the carousel instance

onMount(async () => {

const module = await import('svelte-carousel');

Carousel = module.default;

});

const handleNextClick = () => {

carousel.goToNext()

}

</script>

1. **Collection page – random button to display games**

Using “fHref” and “fSrc” to display final image

<div class="w3-third w3-container w3-margin-bottom">

<a href={fHref[0]}>

<img src={fSrc[0]} alt="elden\_ring" style="" class="disImg w3-hover-opacity">

</a>

</div>

Using “rHref” and “rSrc” to store src and href information

let rHref = [

"/details/gameDetails\_01",

"/details/gD\_genshin",

"/details/gD\_horizon",

...

let rSrc = [

"/gameImages/elden-ring.jpeg",

"/gameImages/genshin\_impact.jpeg",

"/gameImages/horizon\_forbidden\_wests.jpeg",

...

Using function to create a random list and give value to “fHref” and “fSrc”

function onClick () {

const data = [];

for (let i=0;i<maxNum;i++){

data.push(i);

}

const r = Array.apply(null, { length: data.length })

.map((n, i) => i)

.map((n, i, all) => {

const j = i + Math.floor(Math.random() \* (all.length - i));

const v = all[j];

all[j] = n;

return v;

})

.slice(0, maxNum)

.map(i => {

return data[i];

});

for (let i=0;i<maxNumOfPic;i++){

fHref[i]=rHref[r[i]];

fSrc[i]=rSrc[r[i]];

}

}

1. **Details page – href back to collection or go to feed page**

<div class="w3-margin">

<h5>Related Feeds</h5>

<div class="w3-card-4 w3-white w3-round-xxlarge w3-margin-bottom">

<a href="../feeds/feedsDetail\_01"><p class="w3-padding-16 w3-center">Elden Ring Review</p></a>

</div>

</div>

<div class="w3-center">

<a class="w3-round-xxlarge w3-button w3-black w3-section" href="../gallery">

<i class="fa fa-cubes"></i> back to COLLECTION

</a>

</div>

1. **Feed page – every feed is a card**

<!-- Blog entry -->

<div class="w3-round w3-card-4 w3-margin w3-white">

<a href="./feeds/feedsDetail\_01"><img class="w3-round" src="./gameImages/elden-ring.jpeg" alt="Norway" style="width:100%"></a>

<div class="w3-container">

<h3><b>Elden Ring</b></h3>

<h5>Put a ring on it. <span class="w3-opacity">Feb 23, 2022</span></h5>

</div>

<div class="w3-container">

<p>In the 87 hours that it took me to beat Elden Ring, I was put through an absolute wringer of emotion: Anger as I was beaten down by its toughest challenges, exhilaration when I finally overcame them, and a fair amount of sorrow for the mountains of exp I lost along the way to some of the toughest boss encounters FromSoftware has ever conceived.</p>

<div class="w3-row">

<div class="w3-col m8 s12">

<a href="./feeds/feedsDetail\_01" class="w3-button w3-padding-large w3-white w3-border">READ MORE »</a>

</div>

<div class="w3-col m4 w3-hide-small">

<p><span class="w3-padding-large w3-right"><b>Comments  </b> <span class="w3-badge">2</span></span></p>

</div>

</div>

</div>

1. **Feed detail page – normal blog**

No special code

1. **Moment page – click to focus on an image**

<!-- Modal for full size images on click-->

<div id="modal01" class="w3-modal w3-black" style="padding-top:0" alt="" on:click={onClickClose}>

<span class="w3-button w3-black w3-xlarge w3-display-topright">×</span>

<div class="w3-modal-content w3-animate-zoom w3-center w3-transparent w3-padding-64">

<img src={useSrc} id="img01" alt="empty" class="w3-image">

<p id="caption"></p>

</div>

</div>

Use array to storage src information

// Modal Image Gallery

function onClick(indexNum) {

useSrc = rSrc[indexNum];

document.getElementById("modal01").style.display = "block";

var captionText = document.getElementById("caption");

captionText.innerHTML = rAlt[indexNum];

}

function onClickClose() {

document.getElementById("modal01").style.display = "none";

}

1. **Website Guide – a lot of hrefs**

Left side bar

Hover to display different words

No special codes

1. **About page – supabase database, svelte-forms**

<Form on:submit={handleSubmit}>

<input

class="w3-input w3-border"

placeholder="Name"

id="name"

name="name"

on:change={handleChange}

bind:value={$form.name}

/>

...

Return apiResult

{#if apiResult != null}

{#if apiResult == true}

<h2 style="color: white; margin-bottom: -30px;">We've received! <br/>Thanks for your comments!</h2>

{:else}

<h2 style="color: white; margin-bottom: -30px;">Ops! Something wrong with your submit.</h2>

{/if}

{/if}