一、vue预加载问题

1在vue.config.js的配置

const PrerenderSPAPlugin = require('prerender-spa-plugin')

const Renderer = PrerenderSPAPlugin.PuppeteerRenderer

configureWebpack: config => {

        config.plugins.push(new PrerenderSPAPlugin({

            staticDir: path.join(\_\_dirname, `dist`),

            routes: [`/`, `/meetDetail`],

            renderer: new Renderer({

                inject: {

                    foo: 'bar'

                },

                headless: false,

                renderAfterDocumentEvent: 'render-event',

            })

        }))

}

注意：prerender-spa-plugin需要使用cnpm进行安装

2在main.ts的配置

new Vue({

    router,

    store,

    render: h => h(App),

    mounted() {

        document.dispatchEvent(new Event('render-event'));

    }

}).$mount('#app')

3在路由守卫的配置

router.beforeEach(async (to, from, next) => {

    if (to.meta.content) {

        let head = document.getElementsByTagName('head');

        let meta = document.createElement('meta');

        // @ts-ignore

        document.querySelector('meta[name="keywords"]').setAttribute('content', to.meta.content.keywords)

        // @ts-ignore

        document.querySelector('meta[name="description"]').setAttribute('content', to.meta.content.description)

        meta.content = to.meta.content;

        head[0].appendChild(meta)

}

next()

})

4在路由的配置

export default [

    {

        path: "/meetDetail",

        name: "meetDetail",

        component: () => import("@/views/Index/MeetDetail.vue"),

        meta: {

            title: '详情',

            content: {

                keywords: 'aaaaaaaaaaaa',

                description: 'bbbbbbbbbbbb'

            }

        }

    }

]

5在nginx中的配置

location /meeting {

    // 静态化页面重定向路由

    rewrite ^/meeting/meetDetail.\* /meeting/meetDetail/index.html break;

    root D:/pro/portal\_client/vue\_portal\_meeting;

    index index.html;

    try\_files $uri $uri/ /meeting/index.html;

}

二、我js中遇到的问题

1. IE使用dayjs转换后生成无效时间问题

dayjs(data.endTime + ":00").$d

2 includes用法问题

let a = 0（含类型）

[0,1,3].includes(a) true

注意：类型要对上

3 防止滚动

    // 防止滚动

    private stopBodyScroll(isFixed: boolean) {

        let bodyEl = document.body;

        if (isFixed) {

            this.scrollY = window.scrollY

            // @ts-ignore

            document.body.style = `position:fixed;top:-${this.scrollY}px;height: 100%;width:100%;overflow: hidden;`

        } else {

            // @ts-ignore

            document.body.style = ''

            window.scrollTo(0, this.scrollY)

        }

    }

三、h5的app通过打开浏览器跳appstore

    var ua = navigator.userAgent.toLowerCase();

    var isWeixin = ua.indexOf('micromessenger') != -1;

    var isAndroid = ua.indexOf('android') != -1;

    var isIos = (ua.indexOf('iphone') != -1) || (ua.indexOf('ipad') != -1);

    var d = new Date();

    var t0 = d.getTime();

    var openApp = function () {

        var localUrl = createScheme();

        var openIframe = createIframe();

        if (isIos()) {

            //判断是否是ios,具体的判断函数自行百度

            window.location.href = localUrl;

            var loadDateTime = Date.now();

            setTimeout(function () {

                var timeOutDateTime = Date.now();

                if (timeOutDateTime - loadDateTime < 1000) {

                    window.location.href = "你的下载页面";

                }

            }, 25);

        } else if (isAndroid()) {

            //判断是否是android，具体的判断函数自行百度

            if (isChrome()) {

                //chrome浏览器用iframe打不开得直接去打开，算一个坑

                window.location.href = localUrl;

            } else {

                //抛出你的scheme

                openIframe.src = localUrl;

            }

            setTimeout(function () {

                window.location.href = "你的下载页面";

            }, 500);

        } else {

            //主要是给winphone的用户准备的,实际都没测过，现在winphone不好找啊

            openIframe.src = localUrl;

            setTimeout(function () {

                window.location.href = "你的下载页面";

            }, 500);

        }

    }