

# day07网站前台-吐槽与问答

## 学习目标

- 完成吐槽列表与详细页
- 完成发吐槽与评论功能，掌握富文本编辑器的使用
- 完成问答频道功能
- 掌握DataURL和阿里云OSS

## 第一章-吐槽模块

### 1.吐槽列表页

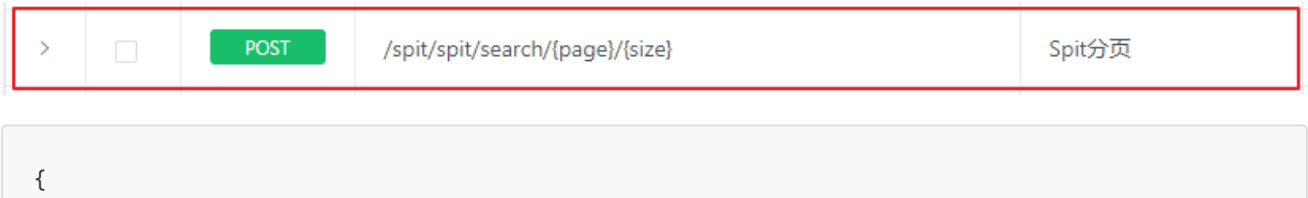
#### 1.1需求



#### 1.2吐槽列表页数据渲染

吐槽列表页已经构建，我们现在来实现数据的渲染

(1) easyMock模拟数据



```

"code": 20000,
"flag": true,
"message": "查询成功",
"data": {
  "total": "@integer(60, 100)",
  "rows|10": [{
    "id": "@integer(1, 1000)",
    "content": "@cword(100,300)",
    "publishtime": "@datetime",
    "userid": "@string",
    "nickname": "小雅",
    "visits": "@integer(60, 100)",
    "thumbup": "@integer(60, 100)",
    "share": "@integer(60, 100)",
    "comment": "@integer(60, 100)",
    "state": "@string",
    "parentid": "@string"
  }]
}
}

```

( 2 ) api目录下创建spit.js

```

import request from '@utils/request'
const group_name = 'spit'
const api_name = 'spit'
export default {
  search(page, size, searchMap) {
    return request({
      url: `/${group_name}/${api_name}/search/${page}/${size}`,
      method: 'post',
      data: searchMap
    })
  }
}

```

( 3 ) 修改pages/spit/index.vue

```

<template>
  <div>
    <div class="wrapper tag-item">
      <div class="fl left-list">
        <div class="tc-data-list">
          <div class="tc-list">
            <ul class="detail-list">
              <li class="qa-item" v-for="(item,index) in items" :key="index">
                <div class="fl record">
                  <div class="number">
                    <div class="border useful">

                        <p class="usenum"><a href="#" class="zan"><i class="fa fa-thumbs-up " aria-

```

```

hidden="true"></i></a></p>
    <p class="zanum"> {{item.thumbup}} </p>
</div>
<div class="border answer">
    <a href="#" class="star"><i class="fa fa-star-o" aria-hidden="true"></i></a>
</div>
</div>
<div class="info">
    <p class="text"> <a href="/assets/spit-detail.html" target="_blank"> {{item.content}}
</a> </p>
    <div class="other">
        <div class="fl date">
            <span>{{item.publishtime}}</span>
        </div>
        <div class="fr remark">
            <a href="#" data-toggle="modal" data-target="#shareModal" class="share"><i class="fa
fa-share-alt" aria-hidden="true"></i> 分享</a>
            <a href="#" data-toggle="modal" data-target="#remarkModal" class="comment"><i
class="fa fa-commenting" aria-hidden="true"></i> 回复</a>
        </div>
    </div>
</div>
<div class="clearfix"></div> </li>
</ul>
<div class="modal fade" id="shareModal" tabindex="-1" role="dialog" aria-
labelledby="myModallLabel">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
aria-hidden="true">&times;</span></button>
                <h4 class="modal-title" id="myModallLabel">分享到</h4>
            </div>
            <div class="modal-body" style="overflow:hidden">
                <div class="jiathis_style_32x32">
                    <a class="jiathis_button_qzone"></a>
                    <a class="jiathis_button_tsina"></a>
                    <a class="jiathis_button_weixin"></a>
                    <a class="jiathis_button_cqq"></a>
                </div>
            </div>
        </div>
    </div>
</div>
<div class="modal fade" id="remarkModal" tabindex="-1" role="dialog" aria-
labelledby="myModallLabel">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
aria-hidden="true">&times;</span></button>
                <h4 class="modal-title" id="myModallLabel">发表评论</h4>
            </div>
        </div>
    </div>
</div>

```

```

    </div>
    <div class="modal-body">
      <div class="comment">
        <span class="who">匿名评论</span>: 今天入职
        腾讯, 产品岗, 普通非985211大学, 上家是不到五十人创业小公司! 16年毕业! 找内推腾讯给的面试机会, 五轮面试!
        可能是我把运气攒一起了!
      </div>
      <div class="form">
        <textarea class="form-control" rows="5" placeholder="匿名评论"></textarea>
        <div class="remark">
          <button class="sui-btn btn-info">发表</button>
        </div>
      </div>
    </div>
  </div>
  <div class="fl right-tag">
    <div class="block-btn">
      <p>来个匿名吐槽, 发泄一下你心中的怒火吧! </p>
      <a class="sui-btn btn-block btn-share" href="~/assets/spit-submit.html" target="_blank">发
      吐槽</a>
    </div>
  </div>
  <div class="clearfix"></div>
</div>
</template>
<script>
import '~/assets/css/page-sj-spit-index.css'
import spitApi from '@api/spit'
export default {
  asyncData(){
    return spitApi.search(1,10, {state:'1'}) .then( res=> {
      return {items:res.data.data.rows}
    })
  }
}
</script>

```

## 1.3 吐槽列表瀑布流

<https://www.npmjs.com/package/vue-infinite-scroll>

修改页面pages/spit/index.vue

```
<div class="tc-list" v-infinite-scroll="loadMore" >
```

修改pages/spit/index.vue ,添加pageNo用于记录页码 增加方法

```

data(){
  return {
    pageNo: 1
  }
},
methods:{
  loadMore(){
    this.pageNo++
    spitApi.search( this.pageNo,10,{state:'1'} ).then( res=>{
      this.items=this.items.concat( res.data.data.rows )
    })
  }
}
}

```

## 2 吐槽详情页

### 2.1需求

点击吐槽的item, 根据id获得吐槽详情



### 2.2 构建吐槽详情页

(1) 根据==spit-detail.html==创建pages/spit/\_id.vue

```
<template>
```

```

<div class="wrapper tc-detail">
  <div class="fl left-list">
    <div class="tc-detail">
      <!-- 标题区 -->
      <div class="detail-tit">
        <div class="detail-author">
          <a href="javascript:;">luckness</a> 发布
        </div>
        <div class="detail-content">
          <p>今天入职腾讯，产品岗，普通非985211大学，上家是不到五十人创业小公司！16年毕业！找内推腾讯给的面试机会，五轮面试！可能是我把运气攒一起了！</p>
          
        </div>
        <div class="detail-tool">
          <ul>
            <li><span class="star"><i class="fa fa-thumbs-o-up" aria-hidden="true"></i>
53</span></li>
            <li><a href="#" data-toggle="modal" data-target="#shareModal"><i class="fa fa-
share-alt" aria-hidden="true"></i> 44</a></li>
            <li><a data-toggle="modal" data-target="#remarkModal"><i class="fa fa-commenting"
aria-hidden="true"></i> 666</a></li>
          </ul>
        </div>
      </div>
    <!-- 评论区 -->
    <div class="comment-area">
      <div class="comment-tit">
        <span>评论</span>
      </div>
      <ul class="comment-list">
        <li>
          <div class="item-photo">
            
          </div>
          <div class="item-content">
            <p class="author"><a href="javascript:;">luckness</a> 发布</p>
            <p class="content">运气+实力吧？不能谦虚哦，沙发！</p>
          </div>
          <div class="item-thumb">
            <div>
              <i class="fa fa-thumbs-o-up" aria-hidden="true"></i> 52
            </div>
          </div>
        </li>
        <li>
          <div class="item-photo">
            
          </div>
          <div class="item-content">
            <p class="author"><a href="javascript:;">luckness</a> 发布</p>
            <p class="content">运气+实力吧？不能谦虚哦，沙发！</p>
          </div>
          <div class="item-thumb">
            <div>

```

```
<i class="fa fa-thumbs-o-up" aria-hidden="true"></i> 52
    </div>
</div> </li>
<li>
    <div class="item-photo">
        
    </div>
    <div class="item-content">
        <p class="author"><a href="javascript:;">luckness</a> 发布</p>
        <p class="content">内推我吧，条件一样，还要人么？内推我吧，条件一样，还要人么？内推我
吧，条件一样，还要人么？内推我吧，条件一样，还要人么？内推我吧，条件一样，还要人么？内推我
吧，条件一样，还要人么？内推我吧，条件一样，还要人么？内推我吧，条件一样，还要人么？内推我
吧，条件一样，还要人么？内推我吧，条件一样，还要人么？内推我吧，条件一样，还要人么？</p>
    </div>
    <div class="item-thumb">
        <div>
            <i class="fa fa-thumbs-o-up" aria-hidden="true"></i> 52
        </div>
    </div> </li>
<li>
    <div class="item-photo">
        
    </div>
    <div class="item-content">
        <p class="author"><a href="javascript:;">luckness</a> 发布</p>
        <p class="content">运气+实力吧？不能谦虚哦，沙发！</p>
    </div>
    <div class="item-thumb">
        <div>
            <i class="fa fa-thumbs-o-up" aria-hidden="true"></i> 52
        </div>
    </div> </li>
<li>
    <div class="item-photo">
        
    </div>
    <div class="item-content">
        <p class="author"><a href="javascript:;">luckness</a> 发布</p>
        <p class="content">运气+实力吧？不能谦虚哦，沙发！</p>
    </div>
    <div class="item-thumb">
        <div>
            <i class="fa fa-thumbs-o-up" aria-hidden="true"></i> 52
        </div>
    </div> </li>
<li>
    <div class="item-photo">
        
    </div>
    <div class="item-content">
        <p class="author"><a href="javascript:;">luckness</a> 发布</p>
        <p class="content">运气+实力吧？不能谦虚哦，沙发！</p>
    </div>
    <div class="item-thumb">
```

```

        <div>
            <i class="fa fa-thumbs-o-up" aria-hidden="true"></i> 52
        </div>
    </div> </li>
</ul>
</div>
</div>
<div class="fl right-tag">
    <div class="block-btn">
        <p>来个匿名吐槽，发泄一下你心中的怒火吧！</p>
        <a class="sui-btn btn-block btn-share" href="/assets/spit-submit.html" target="_blank">
发吐槽</a>
    </div>
</div>
<div class="clearfix"></div>
</div>
</template>

<script>

    import '~/assets/css/page-sj-spit-detail.css'
    export default {
    }
</script>

```

(2) 修改pages/spit/index.vue 页面链接

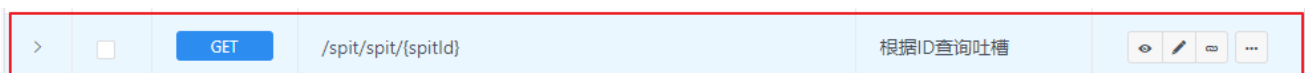
```

<nuxt-link :to="'/spit/'+item.id"><a>{{item.content}}</a></nuxt-link>

```

## 2.3 吐槽详情页数据渲染

(1) easyMock模拟数据



```

{
  "code": 20000,
  "flag": true,
  "message": "@string",
  "data": {
    "id": "@string",
    "nickname": "小雅",
    "content": "@cword(100,300)",
    "publishtime": "@datetime",
    "thumbup": "@integer(60, 100)",
    "share": "@integer(60, 100)",
    "comment": "@integer(60, 100)"
  }
}

```



创建吐槽评论列表的接口: URL: spit/spit/commentlist/{id} Method: GET

>	<input type="checkbox"/>	GET	/spit/spit/commentlist/{id}	根据吐槽id获得评论...
---	--------------------------	-----	-----------------------------	---------------

```
{
  "code": 20000,
  "flag": true,
  "message": "@string",
  "data|10": [{
    "id": "@string",
    "content": "我要评论我要评论我要评论我要评论我要评论我要评论",
    "nickname": "小雅",
    "publishTime": "@datetime",
    "thumbUp": "@integer(60, 100)"
  }]
}
```

(2) 修改api/spit.js

```
findById(id){
  return request({
    url: `/${api_group}/${api_name}/${id}`,
    method: 'get'
  })
},
commentlist(id){
  return request({
    url: `/${api_group}/${api_name}/commentlist/${id}`,
    method: 'get'
  })
}
```

(3) 根据 ==spit-detail.html== 修改pages/spit/\_id.vue

```
<template>
  <div class="wrapper tc-detail">
    <div class="fl left-list">
      <div class="tc-detail">
        <!-- 标题区 -->
        <div class="detail-tit">
          <div class="detail-author">
            <a href="javascript:;">{{pojo.nickname}}</a> 发布
          </div>
          <div class="detail-content">
            <p>{{pojo.content}}</p>
          </div>
          <div class="detail-tool">
            <ul>
              <li><span class="star"><i class="fa fa-thumbs-o-up" aria-hidden="true"></i>
```

```

{{pojo.thumbup}}</span></li>
    <li><a href="#" data-toggle="modal" data-target="#shareModal"><i class="fa fa-share-
alt" aria-hidden="true"></i> {{pojo.share}}</a></li>
    <li><a data-toggle="modal" data-target="#remarkModal"><i class="fa fa-commenting" aria-
hidden="true"></i> {{pojo.comment}}</a></li>
</ul>
</div>
</div>
</div>
<!-- 评论区 -->
<div class="comment-area">
    <div class="comment-tit">
        <span>评论</span>
    </div>
    <ul class="comment-list">
        <li v-for="(item,index) in commentlist" :key="index">
            <div class="item-photo">
                
            </div>
            <div class="item-content">
                <p class="author"><a href="javascript:;">{{item.nickname}}</a> 发布</p>
                <p class="content">{{item.content}}</p>
            </div>
            <div class="item-thumb">
                <div>
                    <i class="fa fa-thumbs-o-up" aria-hidden="true"></i> {{item.thumbup}}
                </div>
            </div> </li>

        </ul>
    </div>
</div>
</div>
<div class="fl right-tag">
    <div class="block-btn">
        <p>来个匿名吐槽，发泄一下你心中的怒火吧！</p>
        <a class="sui-btn btn-block btn-share" href="~/assets/spit-submit.html" target="_blank">发
吐槽</a>
    </div>
</div>
<div class="clearfix"></div>
</div>
</template>
<script>
import '~/assets/css/page-sj-spit-detail.css'
import spitApi from '@api/spit'
import axios from 'axios'
export default {
    asyncData({params}){
        return axios.all( [ spitApi.findById(params.id),spitApi.commentlist(params.id) ]
    ).then(
        axios.spread( function( pojo,commentlist ){
            return {

                pojo: pojo.data.data,

```

```

        commentlist: commentlist.data.data
      }
    })
  )
}
</script>

```

## 3 吐槽点赞

### 3.1 基本功能

( 1 ) easyMock模拟数据

>	<input type="checkbox"/>	PUT	/spit/spit/thumbup/{spitId}	吐槽点赞
---	--------------------------	-----	-----------------------------	------

```

{
  "code": 20000,
  "flag": true,
  "message": "执行成功"
}

```

( 2 ) api/spit.js 增加方法

```

thumbup(id) {
  return request({
    url: `/${api_group}/${api_name}/thumbup/${id}`,
    method: 'put'
  })
}

```

( 3 ) 修改pages/spit/index.vue

```

methods: {
  thumbup(index){
    spitApi.thumbup(this.items[index].id).then( res=>{
      if(res.data.flag){
        this.items[index].thumbup++
      }
    })
  }
}

```

( 4 ) 点赞调用

```

<p class="usenum" @click="thumbup(index)"><a href="#" class="zan"><i class="fa fa-thumbs-up "

```

```
aria-hidden="true"></i></a></p>
```

## 3.2 样式处理

实现点赞后，大拇指图标变色。样式表已经写好，在i标签设置class属性为color就好了

```
<i class="fa fa-thumbs-up color" aria-hidden="true"></i></a></p><p class="zanum"> {{item.thumbup}} </p>
```

(1) 修改pages/spit/index.vue 的代码部分

```
import '~/assets/css/page-sj-spit-index.css'
import spitApi from '@api/spit'
export default {
  asyncData() {
    return spitApi.search(1, 10, {state: '1'}) .then( res=> {
      let tmp= res.data.data.rows.map( item=>{
        return {
          ...item,
          zan: ''
        }
      })
      return {items:tmp}
    })
  },
  data() {
    return {
      pageNo: 1
    }
  },
  methods: {
    loadMore() {
      this.pageNo++
      spitApi.search( this.pageNo, 10, {state: '1'}) .then( res=> {
        let tmp= res.data.data.rows.map( item=> {
          return {
            ...item,
            zan: ''
          }
        })
        this.items=this.items.concat( tmp )
      })
    },
    thumbup(index) {
      this.items[index].zan='color'
      spitApi.thumbup(this.items[index].id).then( res=> {
        if(res.data.flag){
          this.items[index].thumbup++
        }
      })
    }
  }
}
```

( 2 ) 修改pages/spit/index.vue的html部分

```
<a href="#" class="zan">
  <i :class="'fa fa-thumbs-up '+item.zan" aria-hidden="true"></i>
</a>
```

### 3.3 判断是否登陆

要求：点赞必须要在用户登陆的情况下执行，非登陆状态下不能点赞。并且不可重复点赞

导入getUser

```
import {getUser} from '@/utils/auth'
```

修改thumbup ( 点赞 ) 方法

```
thumbup(index){
  if(getUser().name===undefined){
    this.$message({
      message: '必须登陆才可以点赞哦~',
      type: "warning"
    })
    return
  }
  if( this.items[index].zan==='color'){
    this.$message({
      message: '不可以重复点赞哦~',
      type: "warning"
    })
    return
  }
  this.items[index].zan='color'
  spitApi.thumbup(this.items[index].id).then( res=>{
    if(res.data.flag){
      this.items[index].thumbup++
    }
  })
}
```

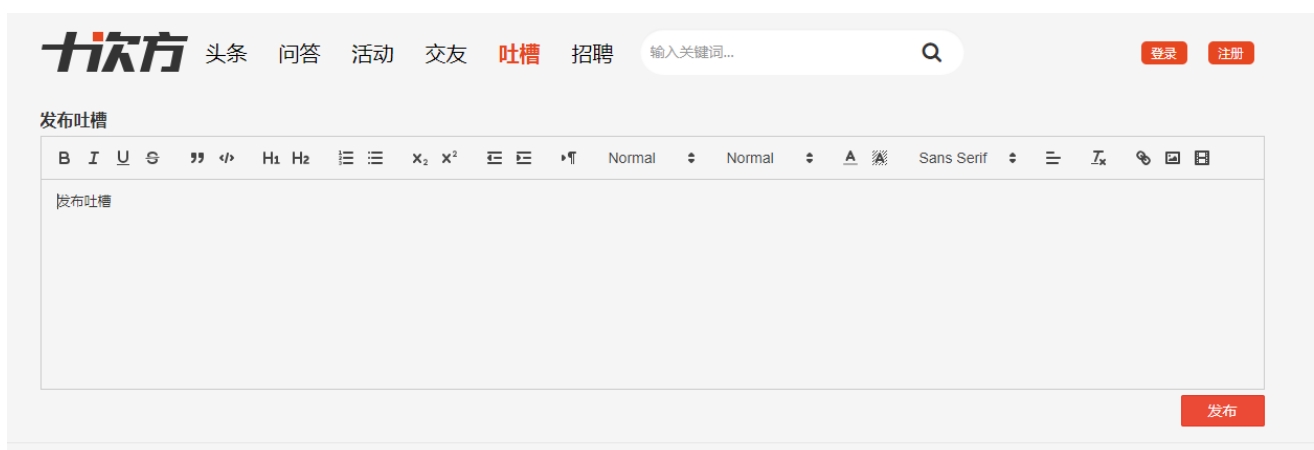
### 3.4 提交token

修改utils/request.js ,每次请求将token添加到header里

```
import axios from 'axios'
import {getUser} from '@/utils/auth'
// 创建axios实例
const service = axios.create({
  baseURL: 'http://192.168.184.133:7300/mock/5af314a4c612520d0d7650c7', // api的base_url
  timeout: 30000, // 请求超时时间
  headers: { 'Authorization': 'Bearer '+getUser().token }
})
export default service
```

## 4.发吐槽

### 4.1需求



### 4.2 构建页面

我们这里用到VUE常用的富文本编辑器vue-quill-editor

详见文档：<https://www.npmjs.com/package/vue-quill-editor>

github教程：<https://github.com/surmon-china/vue-quill-editor/tree/master/examples/nuxt-ssr-example>

( 1 ) 安装vue-quill-editor

```
cnpm install vue-quill-editor --save
```

( 2 ) plugins下创建nuxt-quill-plugin.js

```
import Vue from 'vue'
import VueQuillEditor from 'vue-quill-editor/dist/ssr'
Vue.use(VueQuillEditor)
```

( 3 ) 修改nuxt.config.js ,配置插件和样式

```

plugins: [
  { src: '~/plugins/nuxt-quill-plugin.js', ssr: false }
],
// some nuxt config...
css: [
  'quill/dist/quill.snow.css',
  'quill/dist/quill.bubble.css',
  'quill/dist/quill.core.css'
],

```

(4) 根据静态原型==spit-submit.html== 创建pages/spit/submit.vue

```

<template>
  <div class="wrapper release-tc">
    <div class="release-box">
      <h3>发布吐槽</h3>
      <div class="editor">
        <div class="quill-editor"
          :content="content"
          @change="onEditorChange($event)"
          @blur="onEditorBlur($event)"
          @focus="onEditorFocus($event)"
          @ready="onEditorReady($event)"
          v-quill:myQuillEditor="editorOption">
        </div>

        <div class="btns">
          <button class="sui-btn btn-danger btn-release">发布</button>
        </div>
        <div class="clearfix"></div>
      </div>
    </div>
    <div class="clearfix"></div>
  </div>
</template>
<script>
import '~/assets/css/page-sj-spit-submit.css'
export default {
  data () {
    return {
      content: '',
      editorOption: {
        // some quill options
        modules: {
          toolbar: [
            [{ 'size': ['small', false, 'large'] }],
            ['bold', 'italic'],
            [{ 'list': 'ordered' }, { 'list': 'bullet' }],
            ['link', 'image']
          ]
        }
      }
    }
  }
}

```

```

    }
  },
  mounted() {
    console.log('app init, my quill insrance object is:', this.myQuillEditor)
    /*setTimeout(() => {
      this.content = 'i am changed'
    }, 3000)*/
  },
  methods: {
    onEditorBlur(editor) {
      console.log('editor blur!', editor)
    },
    onEditorFocus(editor) {
      console.log('editor focus!', editor)
    },
    onEditorReady(editor) {
      console.log('editor ready!', editor)
    },
    onEditorChange({ editor, html, text }) {
      console.log('editor change!', editor, html, text)
      this.content = html
    }
  }
}
</script>
<style>
.quill-editor {
  min-height: 200px;
  max-height: 400px;
  overflow-y: auto;
}
</style>

```

( 5 ) 修改pages/spit/index.vue 链接到此页面

```
<router-link class="sui-btn btn-block btn-share" to="/spit/submit">发吐槽</router-link>
```

## 4.3 提交吐槽

( 1 ) easyMock模拟数据

>	<input type="checkbox"/>	POST	/spit/spit	增加吐槽
---	--------------------------	------	------------	------

```

{
  "code": 20000,
  "flag": true,
  "message": "执行成功"
}

```



( 2 ) 修改api/spit.js , 增加提交吐槽的方法

```
save(pojo) {  
  return request({  
    url: `/${group_name}/${api_name}`,  
    method: 'post',  
    data: pojo  
  })  
}
```

( 2 ) 修改pages/spit/submit.vue 引入API

```
import spitApi from '@api/spit'
```

在methods增加方法

```
save(){  
  spitApi.save({ content:this.content } ).then(res=>{  
    this.$message({  
      message: res.data.message,  
      type: (res.data.flag?'success':'error')  
    })  
    if(res.data.flag){  
      this.$router.push('/spit')  
    }  
  })  
}
```

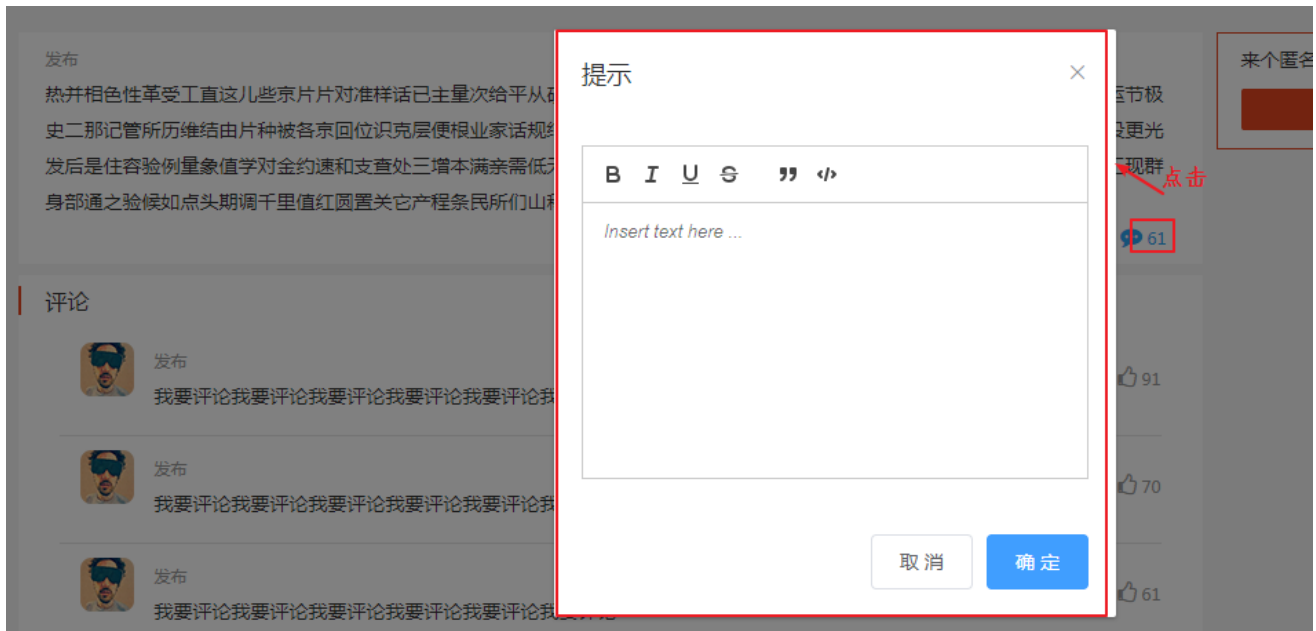
\$router.push()的作用是路由跳转。

( 3 ) 发布按钮调用方法

```
<button class="sui-btn btn-danger btn-release" @click="save">发布</button>
```

## 5.吐槽评论

### 5.1需求



## 5.2 评论弹出框

我们这里的评论弹出框使用elementUI的弹出框来实现

( 1 ) 修改pages/spit/\_id.vue ,添加弹出框, 弹出框中放置富文本编辑器

```
<el-dialog
  title="评论"
  :visible.sync="dialogVisible"
  width="40%" >
  <div class="quill-editor"
    :content="content"
    @change="onEditorChange($event)"
    v-quill:myQuillEditor="editorOption">
  </div>
  <span slot="footer" class="dialog-footer">
    <el-button @click="dialogVisible = false">取 消</el-button>
    <el-button type="primary" @click="dialogVisible = false">确 定</el-button>
  </span>
</el-dialog>
```

为富文本编辑框添加样式：

```
<style>
.quill-editor {
  min-height: 200px;
  max-height: 400px;
  overflow-y: auto;
}
</style>
```

( 2 ) 修改pages/spit/index.vue代码部分

```

data(){
  return {
    dialogVisible: false,
    content: '',
    editorOption: {
      // some quill options
      modules: {
        toolbar: [
          [{ 'size': ['small', false, 'large'] }],
          ['bold', 'italic'],
          [{ 'list': 'ordered'}, { 'list': 'bullet' }],
          ['link', 'image']
        ]
      }
    }
  },
  methods:{
    onEditorChange({ editor, html, text }) {
      console.log('editor change!', editor, html, text)
      this.content = html
    }
  }
}

```

( 3 ) 修改pages/spit/\_id.vue 中的回复链接

```

<a @click="dialogVisible=true;content=''"><i class="fa fa-commenting" aria-hidden="true"></i>
{{pojo.comment}}</a>

```

## 5.3 提交评论

修改pages/spit/\_id.vue , 增加提交回复的方法

```

save(){
  spitApi.save({ content:this.content,parentid: this.pojo.id } ).then(res=>{
    this.$message({
      message: res.data.message,
      type: (res.data.flag?'success':'error')
    })
    if(res.data.flag){
      this.dialogVisible=false
      spitApi.commentlist(this.pojo.id).then(res=>{
        this.commentlist=res.data.data
      })
    }
  })
}

```

编辑提交按钮

```
<el-button type="primary" @click="save">提交</el-button>
```

## 第二章-问答频道

### 1.嵌套布局与标签导航



#### 1.1 嵌套布局

(1) 创建pages/qa.vue

```
<template>
  <div>
    <div class="tab-nav">
      <div class="wrapper">
        <ul class="fl sui-nav nav-tabs navbar-dark">
          <li class="active"><a href="#index" data-toggle="tab">首页</a></li>
          <li><a href="#php" data-toggle="tab">Php</a></li>
          <li><a href="#js" data-toggle="tab">Javascript </a></li>
          <li><a href="#python" data-toggle="tab">Python</a></li>
          <li><a href="#java" data-toggle="tab">Java</a></li>
        </ul>
        <span class="fr more"><a href="./qa-allTag.html" target="_blank">更多</a></span>
        <div class="clearfix"></div>
      </div>
    </div>
    <nuxt-child/>
  </div>
</template>
<script>
import '~/assets/css/page-sj-qa-logged.css'
export default {
```

```

}
</script>

```

(2) 创建pages/qa/label/\_label.vue

```

<template>
  <div class="wrapper qa-content">
    <div class="fl left-list">
      <div class="tab-content">
        <div id="index" class="tab-pane active">
          <div class="tab-bottom-line">
            <ul class="sui-nav nav-tabs">
              <li class="active"><a href="#new" data-toggle="tab">最新回答</a></li>
              <li><a href="#hot" data-toggle="tab">热门回答</a></li>
              <li><a href="#wait" data-toggle="tab">等待回答</a></li>
            </ul>
            <div class="qa-list">
              <div class="tab-content">
                <div id="new" class="tab-pane active">
                  <ul class="detail-list">
                    <li class="qa-item">
                      <div class="fl record">
                        <div class="number">
                          <div class="border useful">
                            <p class="usenum">12</p>
                            <p>有用</p>
                          </div>
                        <div class="border answer">
                          <p class="ansnum">9</p>
                          <p>回答</p>
                        </div>
                      </div>
                      <div class="fl info">
                        <div class="question">
                          <p class="author"><span class="name">luckness</span><span>3</span>分钟前回答</p>
                          <p class="title"><a href="./qa-detail.html" target="_blank">有关PHP初级进阶的问题？
                        </div>
                      </div>
                    </li>
                    <div class="other">
                      <ul class="fl sui-tag">
                        <li>Php</li>
                        <li>Javascript</li>
                      </ul>
                      <div class="fr brower">
                        <p>浏览量 50 | 2017-07-05 15:09 来自 <a href="#">毕鹏 </a></p>
                      </div>
                    </div>
                  </div>
                <div class="clearfix"></div> </li>
                <li class="qa-item">
                  <div class="fl record">

```

```

<div class="number">
  <div class="border useful">
    <p class="usenum">12</p>
    <p>有用</p>
  </div>
  <div class="border answer">
    <p class="ansnum">9</p>
    <p>回答</p>
  </div>
</div>
</div>
<div class="fl info">
  <div class="question">
    <p class="author"><span class="name">牛奶糖</span><span>3</span>分钟前回答</p>
    <p class="title"><a href="./qa-detail.html" target="_blank">springMVC的controller
接收json数据失败</a></p>
  </div>
  <div class="other">
    <ul class="fl sui-tag">
      <li>Php</li>
      <li>Javascript</li>
    </ul>
    <div class="fr brower">
      <p>浏览量 50 | 2017-07-05 15:09 来自 <a href="#">毕鹏 </a></p>
    </div>
  </div>
</div>
<div class="clearfix"></div> </li>
<li class="qa-item">
  <div class="fl record">
    <div class="number">
      <div class="border useful">
        <p class="usenum">12</p>
        <p>有用</p>
      </div>
      <div class="border answer">
        <p class="ansnum">9</p>
        <p>回答</p>
      </div>
    </div>
  </div>
  <div class="fl info">
    <div class="question">
      <p class="author"><span class="name">大白兔</span><span>3</span>分钟前回答</p>
      <p class="title"><a href="./qa-detail.html" target="_blank">监听器中timer查询数据库
</a></p>
    </div>
    <div class="other">
      <ul class="fl sui-tag">
        <li>Php</li>
        <li>Javascript</li>
      </ul>
      <div class="fr brower">

```

```

        <p>浏览量 50 | 2017-07-05 15:09 来自 <a href="#">毕鹏 </a></p>
    </div>
</div>
</div>
<div class="clearfix"></div> </li>
<li class="qa-item">
    <div class="fl record">
        <div class="number">
            <div class="border useful">
                <p class="usenum">34</p>
                <p>有用</p>
            </div>
            <div class="border answer">
                <p class="ansnum">9</p>
                <p>回答</p>
            </div>
        </div>
        <div class="fl info">
            <div class="question">
                <p class="author"><span class="name">luckness</span><span>3</span>分钟前回答</p>
                <p class="title"><a href="./qa-detail.html" target="_blank">服务器上安装了一个考试系
统ASP.NET,安装完成后访问不了,求助</a></p>
            </div>
            <div class="other">
                <ul class="fl sui-tag">
                    <li>Php</li>
                    <li>Javascript</li>
                </ul>
            </div>
            <div class="fr brower">
                <p>浏览量 50 | 2017-07-05 15:09 来自 <a href="#">毕鹏 </a></p>
            </div>
        </div>
    </div>
<div class="clearfix"></div> </li>
<li class="qa-item">
    <div class="fl record">
        <div class="number">
            <div class="border useful">
                <p class="usenum">12</p>
                <p>有用</p>
            </div>
            <div class="border answer">
                <p class="ansnum">9</p>
                <p>回答</p>
            </div>
        </div>
        <div class="fl info">
            <div class="question">
                <p class="author"><span class="name">牛奶糖</span><span>3</span>分钟前回答</p>
                <p class="title"><a href="./qa-detail.html" target="_blank">springMVC的controller

```

接收json数据失败</a></p>

```
</div>
<div class="other">
  <ul class="fl sui-tag">
    <li>Php</li>
    <li>Javascript</li>
  </ul>
  <div class="fr brower">
    <p>浏览量 50 | 2017-07-05 15:09 来自 <a href="#">毕鹏 </a></p>
  </div>
</div>
</div>
<div class="clearfix"></div> </li>
<li class="qa-item">
  <div class="fl record">
    <div class="number">
      <div class="border useful">
        <p class="usenum">12</p>
        <p>有用</p>
      </div>
      <div class="border answer">
        <p class="ansnum">9</p>
        <p>回答</p>
      </div>
    </div>
    </div>
    </div>
    <div class="fl info">
      <div class="question">
        <p class="author"><span class="name">大白兔</span><span>3</span>分钟前回答</p>
        <p class="title"><a href="./qa-detail.html" target="_blank">监听器中timer查询数据库
      </div>
    </div>
    <div class="other">
      <ul class="fl sui-tag">
        <li>Php</li>
        <li>Javascript</li>
      </ul>
      <div class="fr brower">
        <p>浏览量 50 | 2017-07-05 15:09 来自 <a href="#">毕鹏 </a></p>
      </div>
    </div>
    </div>
    <div class="clearfix"></div> </li>
  </ul>
</div>
<div id="hot" class="tab-pane">
  <p>热门回答</p>
</div>
<div id="wait" class="tab-pane">
  <p>等待回答</p>
</div>
</div>
</div>
```



```

    </div>
  </div>
  <div id="php" class="tab-pane">
    php
  </div>
  <div id="js" class="tab-pane">
    Javascript
  </div>
  <div id="python" class="tab-pane">
    python
  </div>
  <div id="java" class="tab-pane">
    java
  </div>
</div>
<div class="fl right-tag">
  <div class="block-btn">
    <p>今天，有什么好东西要和大家分享么？</p>
    <a class="sui-btn btn-block btn-share" href="./qa-submit.html" target="_blank">发布问题</a>
  </div>
  <div class="hot-tags">
    <div class="head">
      <h3 class="title">热门标签</h3>
    </div>
    <div class="tags">
      <ul class="sui-tag">
        <li>Php</li>
        <li>Javascript</li>
        <li>Gif</li>
        <li>Java</li>
        <li>C#</li>
        <li>iOS</li>
        <li>C++</li>
      </ul>
    </div>
  </div>
</div>
<div class="clearfix"></div>
</div>
</template>

```

( 3 ) 创建pages/qa/index.vue

```
<template>
  <div></div>
</template>
<script>
export default {
  created(){
    this.$router.push('/qa/label/0')
  }
}
</script>
```

## 1.2 标签导航

( 1 ) easyMock模拟数据

标签: /base/label/toplist method: get

>	<input type="checkbox"/>	GET	/base/label/toplist	推荐标签列表
---	--------------------------	-----	---------------------	--------

```
{
  "flag": true,
  "code": 20000,
  "data": [{
    "id": "1",
    "labelname": "JAVA"
  },
  {
    "id": "2",
    "labelname": "PHP"
  },
  {
    "id": "3",
    "labelname": "前端"
  },
  {
    "id": "4",
    "labelname": "Python"
  }
]
```

( 2 ) 编写标签API 创建api/label.js

```
import request from '@/utils/request'
const api_group = 'base'
const api_name = 'label'
export default {
  toplist() {
    return request({
      url: `/${api_group}/${api_name}/toplist`,
      method: 'get'
    })
  }
}
```

( 3 ) 修改pages/qa.vue

```
<template>
  <div>
    <div class="tab-nav ">
      <div class="wrapper">
        <ul class="fl sui-nav nav-tabs navbar-dark">
          <router-link tag="li" to="/qa/label/0" active-class="active" exact ><a> 首页</a>
        </router-link>
          <router-link tag="li" :to="'/qa/label/'+item.id" active-class="active" v-for="
(item,index) in labelList" :key="index" >
            <a>{{item.labelname}} </a>
          </router-link>
        </ul>
        <span class="fr more"><a href="./qa-allTag.html" target="_blank">更多</a></span>
        <div class="clearfix"></div>
      </div>
    </div>
    <nuxt-child/>
  </div>
</template>
<script>
import labelApi from '@/api/label'
export default {
  asyncData () {
    return labelApi.toplist().then((res) => {
      return {labelList: res.data.data }
    })
  }
}
</script>
```

## 2.问答列表

### 2.1 最新,热门,等待问答列表

( 1 ) easy-mock模拟数据

>	<input type="checkbox"/>	GET	/qa/problem/newlist/{label}/{page}/{size}	最新问答列表
>	<input type="checkbox"/>	GET	/qa/problem/hotlist/{label}/{page}/{size}	热门问答列表
>	<input type="checkbox"/>	GET	/qa/problem/waitlist/{label}/{page}/{size}	等待回答列表

```
{
  "code": "@integer(60, 100)",
  "flag": "@boolean",
  "message": "@string",
  "data": {
    "total": "@integer(60, 100)",
    "rows|10": [{
      "id": "@integer(1, 1000)",
      "title": "@cword(20,30)",
      "content": "@string",
      "createtime": "@datetime",
      "updatetime": "@datetime",
      "userid": "@integer(1, 1000)",
      "nickname": "小马",
      "visits": "@integer(60, 100)",
      "thumbup": "@integer(60, 100)",
      "reply": "@integer(60, 100)",
      "solve": "@string",
      "replyname": "小牛",
      "replytime": "@datetime"
    }]
  }
}
```

## ( 2 ) API编写 创建api/problem.js

```
import request from '@utils/request'
const group_name = 'qa'
const api_name = 'problem'
export default {
  list(type,label,page,size){
    return request({
      url: `/${group_name}/${api_name}/${type}/${label}/${page}/${size}`,
      method: 'get'
    })
  }
}
```

## ( 3 ) 修改pages/qa/label/\_label.vue 脚本部分

```
import problemApi from '@api/problem'
import axios from 'axios'
export default {
```

```

asyncData({params}){
  return axios.all([problemApi.list('newlist',params.label,1,10),
    problemApi.list('hotlist',params.label,1,10),
    problemApi.list('waitlist',params.label,1,10)  ] ).then(
    axios.spread(function(newlist,hotlist,waitlist ){
      return {
        newlist:newlist.data.data.rows,
        hotlist:hotlist.data.data.rows,
        waitlist:waitlist.data.data.rows
      }
    })))
}
}

```

( 4 ) 修改pages/qa/label/\_label.vue 模板部分

```

<ul class="detail-list">
  <li class="qa-item" v-for="(item,index) in newlist" :key="index">
    <div class="fl record">
      <div class="number">
        <div class="border useful">
          <p class="usenum">{{item.thumbup}}</p>
          <p>有用</p>
        </div>
        <div class="border answer">
          <p class="ansnum">{{item.reply}}</p>
          <p>回答</p>
        </div>
      </div>
    </div>
    <div class="fl info">
      <div class="question">
        <p class="author"><span class="name">{{item.replyname}}</span><span>
{{item.replytime}}</span>回答</p>
        <p class="title"><a href="./qa-detail.html" target="_blank">{{item.title}}</a></p>
      </div>
      <div class="other">
        <div class="fr brower">
          <p>浏览量 {{item.visits}} | {{item.createtime}} 来自 <a href="#">{{item.nickname}}
</a></p>
        </div>
      </div>
    </div>
    <div class="clearfix"></div> </li>
  </ul>

```

## 2.2 切换样式

( 1 ) 定义属性type ,默认值为new

```
data(){
  return {
    type: 'new'
  }
}
```

( 2 ) 修改选项卡

```
<ul class="sui-nav nav-tabs">
  <li :class="type==='new'?'active':''"><a @click="type='new'">最新回答</a></li>
  <li :class="type==='hot'?'active':''"><a @click="type='hot'">热门回答</a></li>
  <li :class="type==='wait'?'active':''"><a @click="type='wait'">等待回答</a></li>
</ul>
```

( 3 ) 修改div的样式为动态获取

最新回答列表

```
<div id="new" :class="'tab-pane '+(type==='new'?'active':'')">
  .....
</div>
```

热门回答列表

```
<div id="hot" :class="'tab-pane '+(type==='hot'?'active':'')">
```

等待回答列表

```
<div id="wait" :class="'tab-pane '+(type==='wait'?'active':'')">
```

( 4 ) 参照最新问答列表编写热门回答列表与等待回答列表内容

```
<li class="qa-item" v-for="(item,index) in hotlist" :key="index">
  .....
<li class="qa-item" v-for="(item,index) in waitlist" :key="index">
```

## 2.3 问答列表瀑布流【自己实现】

<https://www.npmjs.com/package/vue-infinite-scroll>

( 1 ) 修改pages/qa/label/\_label.vue模板部分

```
<div class="qa-list" v-infinite-scroll="loadMore">
  ....
```

( 2 ) 修改pages/qa/label/\_label.vue脚本部分

```

import problemApi from '@api/problem'
import axios from 'axios'
export default {
  asyncData({params}){
    return axios.all([problemApi.list('newlist',params.label,1,10),
      problemApi.list('hotlist',params.label,1,10),
      problemApi.list('waitlist',params.label,1,10)  ] ).then(
    axios.spread(function(newlist,hotlist,waitlist ){
      return {
        newlist:newlist.data.data.rows,
        hotlist:hotlist.data.data.rows,
        waitlist:waitlist.data.data.rows,
        label:params.label //标签ID , 我们需要记录下来
      }
    })))
  },
  data(){
    return {
      type:'new',
      page_new: 1,//记录最新问题列表的页码
      page_hot: 1,//记录热门问题列表的页码
      page_wait: 1//记录等待回答列表的页码
    }
  },
  methods:{
    loadMore(){
      if(this.type==='new'){
        this.page_new++
        problemApi.list('newlist',this.label,this.page_new,10).then( res=>{
          this.newlist=this.newlist.concat( res.data.data.rows )
        })
      }
      if(this.type==='hot'){
        this.page_hot++
        problemApi.list('hotlist',this.label,this.page_hot,10).then( res=>{
          this.hotlist=this.hotlist.concat( res.data.data.rows )
        })
      }
      if(this.type==='wait'){
        this.page_wait++
        problemApi.list('waitlist',this.label,this.page_wait,10).then( res=>{
          this.waitlist=this.waitlist.concat( res.data.data.rows )
        })
      }
    }
  }
}

```

### 3.问答详细页

学员实现

## 4.发布问题页

---

学员实现。使用富文本编辑器（参见吐槽模块的实现）

## 5.标签列表与关注标签

---

学员实现

# 第三章-图片上传

---

## 1. Data URL

---

[Data URL](#)给了我们一种很巧妙的将图片“嵌入”到HTML中的方法。跟传统的用 `img` 标记将服务器上的图片引用到页面中的方式不一样，在Data URL协议中，图片被转换成[base64](#)编码的字符串形式，并存储在URL中，冠以mime-type。

传统方式：

```

```

这种方式中，`img` 标记的 `src` 属性指定了一个远程服务器上的资源。当网页加载到浏览器中时，浏览器会针对每个外部资源都向服务器发送一次拉取资源请求，占用网络资源。大多数的浏览器都有一个并发请求数不能超过4个的限制。这意味着，如果一个网页里嵌入了过多的外部资源，这些请求会导致整个页面的加载延迟。而使用Data URL技术，图片数据以base64字符串格式嵌入到了页面中，与HTML成为一体，它的形式如下



<img

[illegible]

CcWL+fnTKZ7sAomDnlhUNQzfmdLHPSdmawD6vStEtW/lciDfGqYmPINOAhm7nDgCfCADbf2JrrdTGPDi8L99anAyd6PUPR1s  
BFcPyLF2BpaOfDo1pEy7mNI4ppPZgzAgWy9gG//SffrCUCWdQJatWrf1pubM05owK0sp3/FFjaxIU83oSNXWJAP/MyUCcV5d  
HWTQfuGBB6Aax031VUey1cJv3/ks/X2hxnk4u1xXfRi74zzhP56IQrgOHmMXi7pwLvoGcbCQH6LeZL0cywWz1N+VtYMEaQjN  
y9akx4BBYrnW2xVAFdpROE9im9FALLsoI6kxKSmeapdEaTOUPJzOnExpawN2dmCBo/j6QK3wCx9G0844kZaIElPPEPqawXYP  
zNcCboP9ee00saDFvrPXVZ8Nw2IsVKUvPEPC9hYGM4J5dvGTqA2Kd8JzLVBBYhNIXPVEc3N02L8HpdQLuj1EB4U4PZZ4oTBx  
hPEXxki+5Yc4D2zkft2ITsgx8R5NOzdKfWmVm1GG5nuZOV4dIi1lDAGCdseDAB5W26h+EBTo+tlbIjqihwqnsNt9utFBZG5  
M77wuhJ0AnyqRPuZ1EH4XZWnOvXrJXMotFrLBa9ziBR+LsWcGe+r5SZh2F+vdLsKaeUZwe1bHEzFD1ZzLxDiv9Ruv+MRd1QZ  
WHh2MipV1SoHQae17qifoJzaSNxwY0RvvGaUMq/ErSUyH3tBtbNjPrqZ8rjJHOYDvT7bonb6AYVSCMYh30k45HunvQCvfwc  
6u1lG/ANHgCac2nqmULWxDdeqamNVg6f544Qz/OM2dmFCMBzAmbficcTs1iF13zKB4fgf0/Q+n3UxRnhcs6XIwamgTP9gid2  
pQH05Y08LmKHHuQ4KegYNJnuE20cGDdx+g9M8yAHW3pRbcho9HYQH4IIZm2EV175IGmU/UgwzoPYNBj1P9EJPOAAvs4XOM9  
GyV1K6zGRS/s3Yt08Pmc30D3EfbjQwyqcn9885HIowwPrUwJ6wPqB3HmQQDVfKHZSTwJIBwmqWLEyRS1ynZDbUd46WAMhm6  
yBUtguLJQPM+jMRKDBIHsPsoQ0nVACf1/0Dz+AAbFerOI/d7qjndES0C8AU+2UERRN1SxHSP8CdCpFrjPacfnLJsOpJ+ssYw  
LkoHmJqthRf1QeSYBLqp5w++lSwZG6k799rwaUw6cCz+yK1vMncK4MtDEGcrhYCOB7uP5w3MFRk90g/CjieU1ztBWSiQy/J  
0U1tmlaII9wU0t3C4XoptpsyFzfkGd7P1Z8vzIJHmZIsM13IaCEEhdwsvXmCit67eHH/mnz+/21VxvEQ0o/DSd2YnKgAf4oN  
yJEtcekQqBrENZ0vmDd2IARIEJrM9hy0KWIEOpyd0GOTME0VM3AmGX7HqTYvdhTjs3Rt19vhKdX5uhIYnKfoXaDv29hHUKx  
QA8cv38bNCZFOKMAQm+KqRyxMRBGGDxGbWqfEHAPJM5IH2Ws9SBsZ3BvfBpmjBg8Ib12NHqirKo03YYmMGOJBTSfTkQQJ00  
Cc7fZG1rhXgycs7dmuTWrVtJLPFSrrcRMOWeGmz3vGp9Az7eurSNVeK307x9WRg0v1hHIhB1wPrFA8GolJxj40VJG61+JbLe  
s6qzDEo6XtbGLlusVGipkVFZbIa8X11ToFlk53u47Mh03YHcgQ0bYMS3B2p6IYDu8PRTEgD14+nqYC14WY2Ak5dVpbfbY3nTx  
IylWfIX41S7OpzFzXDj0DjLohkIIeJzQ8R7QsQfUM4QiuBoaVUZ0aegDIiHwyzMLy+mcacFAnODTqH1OI+DwNcfb7PZ0vRUh  
Lxe4FkNi6CRDgCpJiic1Sa306FTRQDYtDEf9LCnGyktR1CxbSJX2h9A03XBgt9t1X6INLffeQ9yntQH0ZcCTpfB1o0nC  
Tg/mSi3Getqn1T2AzJGjuzrdr1LFmZ8gbDHDZ8f7+44WQoe/Ofz+zYegVP2w010XtorLa01b1Z6na0jww1AgwXg3K2UHXb4i  
mJ6JVi7lpXijZX9gMSEK1BzqlhB2yoZHDx2Xw2dtCy5pfeR2GKFHe+8LsDhRcp1LeoPOTVPw4Nc9YaJed3j6EB2F1fZNZ1H  
4fdiVX0S2ngPUOgWcWpVK9h7gMFpei0kyQAIejSHEazYQ3580LVKbyGSNGDiTw1A4ajyekIe0fZ3EfyV0D7jOP8uCGPxxkWR  
3VLJaQBmN0zFGu2ZCovOUqZ01WcwnEeEt2+8nq9FfQPp8vZrHda+ihWqrCnIjfk/f9GXXKcDW3oWVA44I9zJySk125uDDBN/A  
sVwAT+T6vz5WdxH3ZTY00XbKyb+NEw1CF4g98ShYZLz1IhuPu8oEtVWbneT1LPgn++ZmrZy6fx5V1mNYM1LJa5TqVNFu7Xv  
10uVyKzSoiermLb4Wi4Ds3qMnNDM+ClHru/+5HuIarClLuB7t/iNDgoA5ZF6HYXU/xkjse6EBYpp1LcIbPH3aawWAsXn/co/  
mN0i9nd5fdTddGLL13o/tEQnAd0ks9Qugn8W+HfordxAsXB6ti055MSx1jKs1D4Phh04IMH3y+srGPBeie+VcgH6x4wKy+n9  
OB4WKd61a8Qt6FYwpy/iEE9jK/getmU1y6hPHgRwKMEC5PQS4+zDr2Y0vki9MUNPHHge2HoWUvvvCM8T+L2hu8kXKawrXwRp  
dkhpBnN+jq+byDnhfvpXF+835G/Cda7zEj1qsB5mPus8ng9u2TDgU210dwu186eawkpViCHQPMGHuQ5hw8a+fcXUpJTrsIQ0  
RF4It/wD3Fn8NYESwSwYZG3B99jkMGtYwx3BqYyGAvrX7AYTqI4/fCOISzy6KeJ3sj3hSxmjeI83Aypq3kAKAZUL5sGUrwrm  
FtQIyXoh+bi39WFdG/zwAf7bcA66FquM8qFnx0Wa59h/fUwD+zd9KyXkM94pcxptRXfb+Q6Qi5uyVIDwLONgYH4kRiAPAGhP  
DiJfs/tCXEFzZzAU2WeUt5TQwaNjcJcNia14fewXsVxU/nNGj/b/gbEzgn2n07PXtCmghe4PpDvIea8BADBQv18ILxcr2w6  
Do1arFNJ7/P1y1ACpAm8QSHw1nD5XDtIQBVP92skvScSdWrVeddjWUVPXGiwPLR1GiPxKJZp4AbWncb0nd7tqV0qcF6BmYv5  
IsVbQxM3ReqE19vFsStXOYASN+JgQP0kMgAuI7iXPzu+ybq18Ez7EH8piDu/8eC7mUUN4JCR9aBMcvrxwRjySGF0+BdHA0Mn  
zasqmcKK4ARYba+nN4ZzhMB1nKwaE3xXWnACPSCP4u/aAR5H+M37sf5VHAu+uU8H/OpDtZr3DwGymMoXAbG61GUD3/qgQXS  
dEP19CzdAoPc9/Aw2Micy1FmHA6UZq+SKsEzx0/j0HXEu3FHPUHnKxiAw9xnwOkdD8hcx+2vtVzu9wHeqWlnbY/wUgcJ4uAM  
67EA1yvm4a/50BV8k+W4zxwERkXYzEYx02oA2ryoP6c2+kGFr/A1tdQo+/nmfU2SnsJdxLS3sSzPQD4EaULMNCsWf8BuI6w2  
HAzx1VTotzWZaTBuVmirV63Ltah+qBu2J4BP7zGGFw8wk710kPs6Rwm7/163nQt5oF2jH6vEdJ8y9dz0d2tFC6kCJLKuoDCT  
RQAsnOU8Ecj6yfiyFzgIa4n2g5/aGCFKKJx/Z4URF2dfqBn4nkZunfLEuGbAmz9TYxJ59GJdJvJvdmHdHa5301Ffzd4Wzscj  
pqqqn7X0z195JLs7FP216fThw471+12rU1KSm1R0YdaU/Lyyi36g90PrMRfasBAhxs8zPDXsNhwh2Vj3G+mxtbZ3Ac8S25ku  
Rk+cas4Dg6Q+kSEzvuDAXU9z77FnCcG0gHK89tKNp08ACyaHvXjew9iwmJ0uyQCynTcI08TEXSWv3A7duSJo5BFraQI31EiX  
1F7HDGQCpCvfoqT0XqifvxX3TDtYtQ+RvfRcZ5S8c3vu9Hr9R40GKeKqtrtSunJM/pnDN5JIGPy1cAZMGz42GopKT81JSa3k  
Dkir16fV8k7nicCp5vZfTMhaCfPbhN5c1nHoQfrQ09LQkvJ0HfCxZ7iQTJZAJfCSiiMBb1Yt/my69qa46w4cTXcCrHuMj0D6  
1JTDdZqatHv16zqMzaWfMJAHCe6YjKV0Zb1JN14EaquKnN+Mc96FpAnRPTtM0bHGdczAIyvsamIXhIfAv4fn4WNDCL1A7UU  
v0i6MpDnKAlDej9/TIGres9YEDjKuA29w27b1R0jeo1ppD05VKkWhu050GRHUwQ1brEcr6CNMNPdIoLuSO/4cf+gSDWiWD1c  
tp1LNJs4t1zJ90fFtJ+wHnY+L11/Btxq4VBUo1CJN2yWpSfBa4G7/BUyruhqJwrwa3qHvrdzsLua8wDeTW1wTxY89CubH3Td  
UJ3GLEZJJ8nB8MDthvUYAB2F7iQTvpJqWuVU0AhN8AsWzA/v0f/VChnu110V12R+yg2vhAluBM600+9qUOG7PP7/JmBQGdAk  
uy5MT18pg0Z2tZut49zu923UJmJZefIBRdty58tpyl/nPhDyc8/EWau8ZQVf1RL4CgQvdZzp+tuRLDcbReABn1+A1vW1nH00  
hZVVhvKcfqZ3tvDg/dHFu/WSop/TaVsk51MWDK4g979MJzVSp4QKD1MtXDa3Ua/Z7Mx4xYW5aAvbBrWXEEXxBaDp1TWub5vw  
pZdjI1aLArnsjcdTOHdqA4b2BoHo8Fz9Ew/3RUcui/rXmtZd1rD1r2Z3C9d+FtmKDGc/mkpeBhaOT1SUxvTQP3S7XI3Ew8JE  
c+2xrim5xfbXLNKe2T6cOG5wb8/10BLbBFC2hbKc10euHgojlzSsSp/oMG0eCwNSdAtCYd6moCTGun09XU6XC4S0Ci4wSma  
JvxwSRFxUXK0aNHsAEMfzT4Y4vb5QMemG/y4DtMnQST537mTCL9iy1gGwStcVPmhOIA3kd5TFPKb50GUnyoeO0VK0eS4rpUK  
X/+NkzG2Bd0PoswAPM0peJfd9vN8Ye4/I1UfnNe12nDuhbyuZ0erRTEe//WqJ6HvLEndaotxf3GaT+j/LxUVns2FUPc/isDq

```
Bub0gcrZWcBHUYJ57gu4tK73RksXfkdpElj8zjS7KQ0yBfWv6vYpI78J9Ez0V9yt1LxYI+9XE/5AJpVSpTnEzo+qZL0m1fd7
oSRKv5Eti34qmoTsraFQYSHGF0qgyD4yMaF2YLP9PTBS7k8VN1KYdPTK6Xg0YEFN5xffv15Ae1191ZmP0ec4lQZiuqY156pq
S2JQ7xDI1WDMrCUjvSSwV1uo0sl6EBh8CgCaGz10gr3+nMJPmXfxfk5x3L6LZs//51498XprAGPTr3S0oc6nc4pBKQ6pdxHG
uz1uU95EMTCfYjLFBuuFLyYk/vbYxbqNnGK06kFTymI0tMHkZ4yzuV0N9G5RQXuo5cgcR9bGfupwH108JD0pBGn+cHr82YF/
IFn58+eFYh3WZz+F0Ap40RptVXV/ojdrnZ20JzNnA5ncnk0Y477wAjh9f10+HzebwIB/3J/IPDygtmzC+LdFKc/LXgMjAspN
pvanrjJNaqqt1BVWwP6XcOmqtV0GG1YTNS0E5qmHdUCga8DmraJfn+0MCszbgCI011B/xVgA0X9oz13Z9A5AAAAAE1FTkSuQ
mCC">
```

vue-quill-editor的图片上传默认采用Data URL方式。

## 2. 辅助插件vue-quill-editor-upload

如果你不想使用Data URL方式存储图片，我们可以通过一个辅助插件来让vue-quill-editor实现传统方式的上传。

参考地址: <https://github.com/NextBoy/vue-quill-editor-upload>(复制时候注意==uploadConfig==)

(1) 安装：

```
cnpm install vue-quill-editor-upload --save
```

(2) 修改submit.vue 引入插件

```
import {quillRedefine} from 'vue-quill-editor-upload'
```

(3) 将editorOption的值改为{}

```
data () {
  return {
    content: '',
    editorOption:{}//修改此处！
  }
}
```

(4) 新增created 钩子函数

```
created () {
  this.editorOption = quillRedefine(
    {
      // 图片上传的设置
      uploadConfig: {
        action: 'http://localhost:3000/upload', // 必填参数 图片上传地址
        // 必选参数 res是一个函数，函数接收的response为上传成功时服务器返回的数据
        // 你必须把返回的数据中所包含的图片地址 return 回去
        res: (response) => {
          return response.info
        },
        name: 'img' // 图片上传参数名
      }
    }
  )
}
```

```
}
```

### 3. Multer (了解)

---

课程中提供了上传图片的服务端代码 upload-server，我们可以先测试运行后，对照在线文档阅读并理解代码（课程不要求学员独立编写此段代码）

```
cnpm install
npm run start
```

这段代码主要应用两项技术：

- (1) Express --node.js的web框架 在线文档：<http://www.expressjs.com.cn/4x/api.html>
- (2) Multer --Express官方推出的，用于multipart/form-data请求数据处理的中间件 在线文档：<https://github.com/expressjs/multer/blob/master/doc/README-zh-cn.md>

## 4 云存储解决方案-阿里云OSS

---

为了能够解决海量数据存储与弹性扩容，我们在十次方项目中采用云存储的解决方案- 阿里云OSS。

### 4.1 准备工作

- (1) 申请阿里云账号并完成实名认证：可以使用我们之前发短信用云的阿里云账号。
- (2) 开通OSS: 登录阿里云官网。将鼠标移至产品找到并单击对象存储OSS打开OSS产品详情页面。在OSS产品详情页中的单击立即开通。开通服务后，在OSS产品详情页面单击管理控制台直接进入OSS管理控制台界面。您也可以单击位于官网首页右上方菜单栏的控制台，进入阿里云管理控制台首页，然后单击左侧的对象存储OSS菜单进入OSS管理控制台界面。



### (3) 创建存储空间

新建Bucket，命名为tensquare，读写权限为公共读



新建 Bucket

命名：

Bucket 命名规范：

- 只能包含小写字母，数字和短横线
- 必须以小写字母和数字开头和结尾
- 长度限制在 3-63 之间

所属地域：

华北 2

相同地域内的产品内网可以互通；订购后不支持更换地域，请谨慎选择

EndPoint：

oss-cn-beijing.aliyuncs.com

存储类型：

标准存储

- 标准存储类型：高可靠、高可用、高性能，数据会经常被访问到
- 低频访问类型：数据长期存储、较少访问，存储单价低于标准类型
- 归档存储类型：数据长期存储、基本不访问，存储单价低于低频访问型

读写权限：

私有

- 私有：对 Object 的所有访问操作需要进行身份验证
- 公共读：对 Object 写操作需要进行身份验证；可以对 Object 进行匿名读
- 公共读写：所有人都可以对 Object 进行读写操作

## 4.2 代码编写

对象存储 OSS

视频简介

海量、安全、低成本、高可靠的云存储服务，提供99.99999999%的数据可靠性。使用RESTful API 可以在互联网任何位置存储和访问，容量和处理能力弹性扩展，多种存储类型供选择全面优化存储成本。  
OSS“同城冗余存储”公测中，机房级容灾能力，立刻申请 | OSS Select重磅发布，为大数据应用加速

折扣套餐

管理控制台

产品价格

学习路径

文档&SDK

快速开始

功能概述

参考: [https://help.aliyun.com/document\\_detail/32070.html?spm=5176.87240.400427.55.23844614RBUD18](https://help.aliyun.com/document_detail/32070.html?spm=5176.87240.400427.55.23844614RBUD18)

(1) 安装ali-oss

```
cnpm install ali-oss --save
```

(2) 修改upload-server的server.js

```

let OSS = require('ali-oss');

let client = new OSS({
  endpoint: 'oss-cn-shenzhen.aliyuncs.com',
  accessKeyId: 'xxx',
  accessKeySecret: 'xxx',
  bucket: 'tensquare-000'
});

```

```

app.post('/upload', upload.single('img'), (req, res) => {
  // 没有附带文件
  if (!req.file) {
    res.json({ ok: false });
    return;
  }

  // 输出文件信息

  console.log('=====');
  console.log('fieldname: ' + req.file.fieldname);
  console.log('originalname: ' + req.file.originalname);
  console.log('encoding: ' + req.file.encoding);
  console.log('mimetype: ' + req.file.mimetype);
  console.log('size: ' + (req.file.size / 1024).toFixed(2) + 'KB');
  console.log('destination: ' + req.file.destination);
  console.log('filename: ' + req.file.filename);
  console.log('path: ' + req.file.path);

  putStream(req, res);

});

//上传的方法
async function putStream (req, res) {
  try {
    // use 'chunked encoding'
    let stream = fs.createReadStream(req.file.path);
    let result = await client.putStream(req.file.originalname, stream);
    res.json({ ok: true ,info:result.url});

  } catch (e) {
    console.log(e)
  }
}

```

### (3)启动

```
node server-oss.js
```



