**Vtor**

1. **概述**

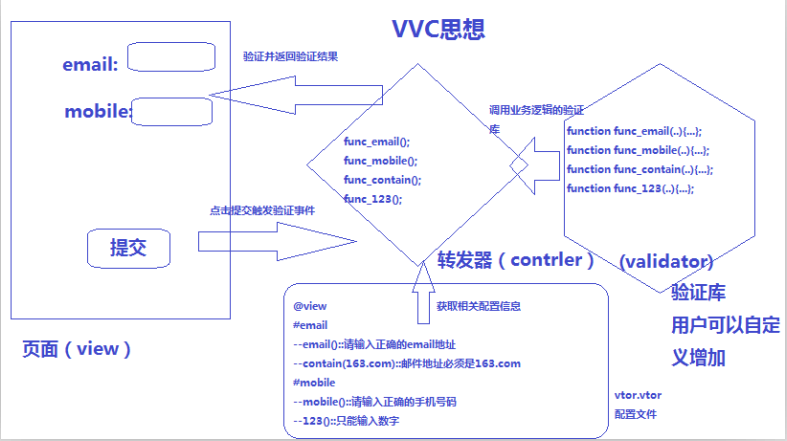
**vtor 是一个全新的解耦合页面验证js框架。根据全新的VVC思想(validator-view-controler). 基于配置，使验证与页面分离，更加方便、高效。**

**Vtor 验证库包含了25种以上的常用验证，email,url,phone,mobile,ajax等等。**

**用户可以自定义添加或覆盖已有的验证函数库；**

**用户可以自定义配置文件组织格式。**

**VVC的工作图：**

****

1. **用法**
   1. **引入框架**

**<script src='js/jquery.js'> </script>**

**<script src='../jquery.vtor.js'> </script>**

* 1. **给需要做验证的DOM元素添加ID属性**

**如：demo.html中，对email验证**

**电子邮箱地址：<input type='text' id='iemail' value=''/>**

**2.3配置验证部件**

**如：在验证文件demo.vtor中,配置对电子邮件进行格式验证且必须是163邮箱的**

**@demo**

**#iemail**

**--email()::邮箱格式不正确**

**--contain(163.com)::请填入163邮箱地址**

**验证文件格式说明：**

**注释方式：**

**多行注释/\* \*/**

**单行注释://**

**@view //页面模块名以@开头；可以有多个**

**#DOM ID //页面上DOM ID以#开头；可以有多个**

**--functionName([param])::message //验证函数类型，参数，提示**

**/\***

**验证语--开头；**

**::后表示验证错误提示信息；无参数时()可省略不写；**

**可以复合验证；既一个DOM可以有多个验证语句；**

**可以有多个。\*/**

**\*/**

**2.4执行验证**

**<script>**

**$(function(){**

**//初始化；参数1：验证文件路径；参数2：页面模块名**

**$vtor.$('demo.vtor',[‘demo’]);**

**$("#submitButton").click(function(){**

**var r=$vtor.$v();//执行验证**

**if(r){**

**$( ‘form’).submit()**

**}**

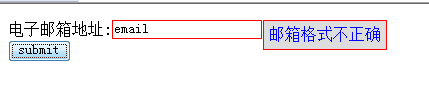
**});**

**})**

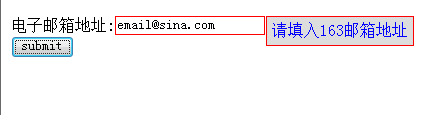
**</script>**

**2.5验证结果**

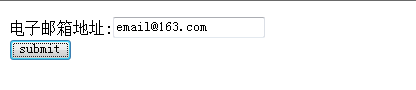
**邮箱格式不正确：**

****

**不是163邮箱：**

****

**全正确：**

****

1. **用户自定义配置**

**3.1概述**

**用户可以通过在vconf/vtor.config.js文件中自定义修改验证文件组织格式、扩充或覆盖验证函数库。**

**需要引入js**

**<script src='../vconf/vtor.config.js'> </script>**

**3.2自定义验证文件组织格式**

**自定义验证文件的代码：**

**custom.vtor={**

**msg\_type:'single'**

**};**

**相关参数配置项如下表：**

|  |  |  |
| --- | --- | --- |
| **参数名** | **参数说明** | **默认** |
| **func\_pre** | **验证函数前缀** | **func** |
| **view\_pre** | **页面模块前缀** | **@** |
| **id\_pre** | **DOM ID前缀** | **#** |
| **validateTerm\_pre** | **验证语句前缀** | **--** |
| **msg\_pre** | **验证提示信息前缀** | **::** |
| **param\_begin** | **验证函数参数开始字符** | **(** |
| **param\_end** | **验证函数参数结束字符** | **)** |
| **param\_split** | **验证函数参数分隔字符** | **,** |
| **msg\_type** | **验证结果信息提示类型:single只提示最靠前的一个错误信息；multiple提示全部错误信息** | **single** |
| **msg\_class** | **提示框的样式** | **vtor-input-err-msg** |
| **errInput\_class** | **输入框错误的样式** | **vtor-input-err-border** |
| **vtor\_suf** | **验证文件扩展名配置** | **vtor** |

**\*以上参数均区分大小写**

**3.3自定义扩充或覆盖验证函数库**

**用户自定义函数的代码：**

**custom.funcs={**

**//只能输入数字（无参数）**

**func\_numeric:function(id){**

**var val=$vtor.$id(id);//获取dom id的输入值**

**return /^\d+$/.test(val);},**

**//只能输入限定位数的数字（有参数）**

**func\_numericLimit:function(id,param){**

**var val=$vtor.$id(id);//获取dom id的输入值**

**if(val.length!=param[0]){return false;}**

**return /^\d+$/.test(val);**

**}**

**}**

**定义好以上两个函数，用户就可以在vtor配置文件中配置使用了**

**例：**

**@demo**

**#inum**

**-- numeric()::请输入数字//无参也可以：-- numeric::请输入数字**

**-- numericLimit(4)::请输入4位数字**

**函数格式= <func\_pre><\_><functionName>:function(<id>[,param]){[functionBody]};**

**既：**

**函数格式=验证函数前缀[默认func]+下划线+函数功能名称:function(DOM ID[,参数]){[方法体]}**