laoduliu

网页版企信

技术架构说明

1. 前端架构requirejs+angularjs+bootstrapUI

本项目是单页面应用，入口文件为WebRoot/index.html

在index.html中引入requirejs：

<script type="text/javascript" src="js/require.js" data-main="js/main.js"></script>

1. Requirejs

Requirejs用于模块化加载和管理页面的js文件，遵循CMD规范，在index.html中引入require.js后，指向js的总入口文件js/main.js。

main.js中主要配置两种信息：1、配置需要引入的js文件paths:（所有js文件无需加js后缀）。2、配置js文件之间的依赖关系shim。

例如：本项目用到了angularjs和jquery，那么对于模块化js来说它们有一个总的输出export，别名分别为angular和jquery。

再例如：validate插件需要依赖jquery，故此添加deps（多个依赖分别逗号隔开）: ['jquery']。

最后main.js对我们的主体js框架angular进行了初始化：



define是使用requirejs定义了一个模块，在这个模块中，引入了以下模块：angular（主体框架）、app(工程主体，后续说明)、router（angular用于控制页面跳转的路由插件）

1. Angularjs

Angularjs是一套前端框架，主要是采用mvvm数据双向绑定的方式构建前端页面（简单来说服务端获取的数据和前台修改的数据能互相同步）。

对于Angularjs网上有很多教程，本项目只需要关心router、controller和templete即可。

app.js主要是通过配置router区分不同的模板页面和对这个模板页面控制的controller（本项目其实只有一个页面，其他都是各路由下的模板）。具体配置参见app.js中的config部分。

注意：这里强调下异步加载的概念：



例如在注册模块中使用require.js的require方法，目的是在加载register.html模板时才加载controller/registerController.js。这样可以提高页面初始化的效率，达到按需加载的效果。其他模块已然。

1. 后台消息推送框架pushlet

使用Pushlet可以让web应用建立长连接实现消息推送。前台使用[WebRoot](https://github.com/laoduliu/QX/tree/master/WebRoot)/[js](https://github.com/laoduliu/QX/tree/master/WebRoot/js)/**ajax-pushlet-client.js实现前台页面的消息接收与推送。**

**前台初始化参见：**/[WebRoot](https://github.com/laoduliu/QX/tree/master/WebRoot)/[js](https://github.com/laoduliu/QX/tree/master/WebRoot/js)/[controller](https://github.com/laoduliu/QX/tree/master/WebRoot/js/controller)/**loginController.js的31行，在人员登录成功后将此人id纳入‘sendMessage’这一个消息推送的主体，之后就能接收发送消息了。**

**前台接收消息调用方法参见**[WebRoot](https://github.com/laoduliu/QX/tree/master/WebRoot)/[js](https://github.com/laoduliu/QX/tree/master/WebRoot/js)/[controller](https://github.com/laoduliu/QX/tree/master/WebRoot/js/controller)/**chatController.js的81行-90行。**

后台使用pushlet提供的jar包或类代码调用Event event = Event.createDataEvent(‘**sendMessage’**’);实现消息的发送（我们主要通过后台来推送消息，前台接收消息），这里我们是往**sendMessage这个主体中定义消息，通过Dispatcher.getInstance().unicast(event, receiveId);来发送给具体的接收人id。**