耗材物资云文档

作者	**lpz**
版本	1.0.0
版权	©2019, 快享医疗.
修订时间	2019-10-11 11:09:10

1 简介

耗材物资云框架

> 这是旧框架，目前除耗材物资云相关项目之外，不建议采用该框架

2 系统初始化

2.1 webconfig参数说明

appSettings参数

> 该部分配置均配置在 `configuration>appSettings` 节点中，形式如下

```
<add key="key" value="value"/>
```

AppVersionFilePath

移动APP版本记录文件

> 指示用以记录移动APP版本信息的文件地址

```
<add key="AppVersionFilePath" value="HCWZ.Download\AppVersion.txt"/>
```

EnableLoginMultiEndPoint

启用单账号多点登录

> 用以指示是否启用单账号多点登录， `true` 为启用，其他值或不配置该节点为不启用

```
<add key="EnableLoginMultiEndPoint" value="true"/>
```

EnableAuditLog

启用审计日志

> 用以指示是否启用审计日志， `true` 为启用，其他值或不配置该节点为不启用；
> 启用后会会将每个请求和每个和DEP平台交互的请求的地址，参数，结果等信息记录在数据库 `AuditLogs` 表中；
> 启用后会增加数据库的压力，有可能导致数据库日志文件增长速度过快；
> 不建议启用该项
> 如果启用该项，请设置数据库日志文件自动收缩

```
<add key="EnableAuditLog" value="false"/>
```

> 历史审计日志会定期删除，只保留70天以内的

ValidateCodeType

配置系统验证码字符类型

> 用以配置系统验证码字符类型，具体配置如下：
> 1:纯数字，2:纯字母，其他或不配置：字母加数字

```
<add key="ValidateCodeType" value="1"/>
```

SiteHost

配置当前站点的域名及地址

> 只有当物资云不作为一级域名发布时使用；
> 物资云作为一级域名发布时一定不要配置该项

```
<add key="SiteHost" value="http://localhost:9070/hcwz/"/>
```

DisableLoginValidCode

登录时是否禁止检查验证码

> 指示登录时是否禁止检查验证码， `true` 为不检查，其他值或不配置该节点为检查；
> 该配置项是提供给自动化测试时使用，其他场景请勿配置

```
<add key="DisableLoginValidCode" value="true"/>
```

锁定策略配置

用户频繁登录错误时对用户或IP进行锁定的策略，用于防止暴力破解密码
该配置分为两部分

节点注册配置

配置路径： `configuration>configSections`
配置内容：

```
<section name="strategies" type="Winning.DMSP.Application.ServiceContracts.LockStrategySection,W
```

配置说明：

> 该配置定义下面详细策略配置的解析类，没有本配置下面的配置不会生效

详细策略配置

配置路径： `configuration>strategies`
配置内容：

```
<add type="IP" timespan="2H" errorcount="20" timespanlock="1D"/>
<add type="User" timespan="2H" errorcount="5" timespanlock="2H"/>
```

配置说明：

> 类型type包含IP和User两种，分别代表同一IP和同一用户，
> timespan是连续的时间段；
> errorcount是连续错误的次数；
> timespanlock是锁定的时长；
> 其中timespan和timespanlock的格式为数字加字母，
> 字母包括S-秒，M-分钟，H-小时，D-天，F-永久
> 以 `<add type="IP" timespan="2H" errorcount="20" timespanlock="1D"/>` 为例，
> 表示同一IP地址在2小时内连续错误达到20次将被锁定1天时间

> 注：1. 超级用户admin不会被锁定，2. IP或User被锁定后，超级管理员admin可在锁定信息页面进行查看和解锁

以下错误情况被计入登录错误次数：

> 用户名或密码不正确；
> 如被系统识别为其他模拟的非法请求，

以下错误情况不被计入登录错误次数：

> 验证码错误；
> 在锁定状态下进行的登录错误

策略判定逻辑：

1. 将所有策略以错误次数从小到大排序
2. 逐一判定策略，若满足锁定条件，执行锁定策略
   当同时有多个策略满足条件，只按第一个满足条件的策略锁定
3. 无满足条件的策略，返回剩余可重试次数

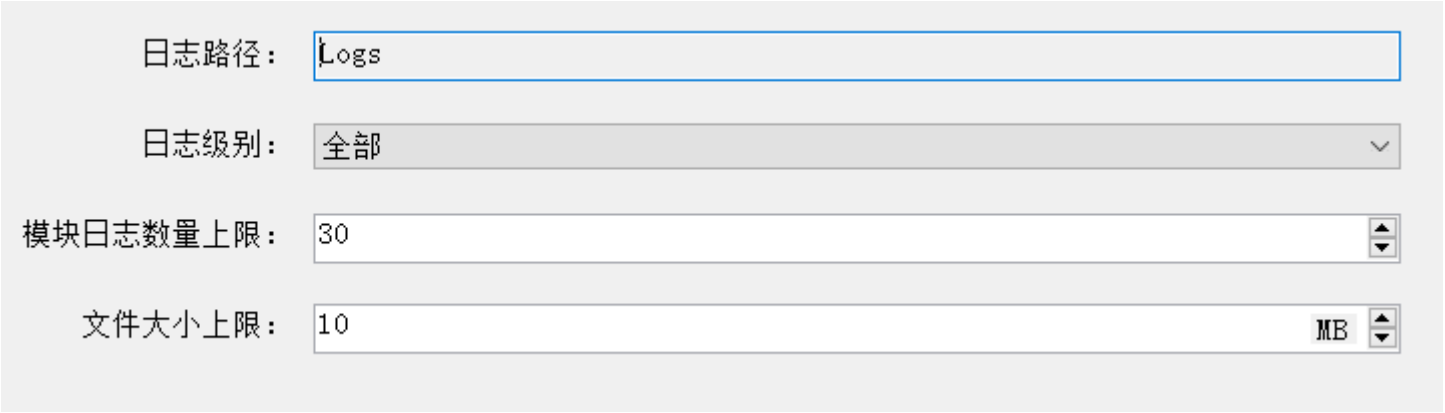
2.2 日志配置

物资云系统的日志输出统一使用log4net模块实现
物资云系统通过日志总线对log4net进行封装，通过配置工具对日志进行配置，实现功能：

1. 可区分模块输出日志
2. 自动创建日志文件
3. 日志文件大小和数量均可限制，超出数量限制会滚动删除最早的日志文件

配置

日志配置界面如下：



日志路径：不可修改，固定为web项目下Logs目录，所有的日志文件均输出在此文件夹内；
日志级别：标识输出日志的级别，包含以下级别：

> 全部：输出所有日志
> 调试：输出所有级别不小于debug的日志
> 信息：输出所有级别不小于info的日志
> 警告：输出所有级别不小于warn的日志
> 错误：输出所有级别不小于error的日志
> 无：不输出任何日志

模块日志数量上限：模块由业务系统定义，每个模块的日志文件数量单独计算；
文件大小上限：单个日志文件的大小上限，单位兆(MB)

日志文件管理

1. 首先日志总线按模块区分文件夹，在单个模块的文件夹中由log4net管理组织日志文件；
2. 按照当前日期命名文件，文件不存在则自动创建，并将日志内容追加到该文件，如：log.2019-09-26.log；
3. 如果文件大小超出限制，将会创建新的log.2019-09-26.log文件，原有的文件则重命名，本例为log.2019-09-26.1.log；文件管理组织方式为滚动日志，如下以模块日志数量上限设置为10为例：

源文件名	操作	新文件名
log.2019-09-26.9.log	删除	
log.2019-09-26.8.log	重命名为	log.2019-09-26.9.log
...	...	...
log.2019-09-26.2.log	重命名为	log.2019-09-26.3.log
log.2019-09-26.1.log	重命名为	log.2019-09-26.2.log
log.2019-09-26.log	重命名为	log.2019-09-26.1.log
	创建新的	log.2019-09-26.log

   同一日期的日志文件中，无序号后缀的文件始终是最新的，序号越大，文件越早。
4. 如果同一模块的日志文件数量超出上限，log4net会自动删除同一模块中最早的日志文件

3 部分功能介绍

3.1 登录流程

登录用例

1. 访问Login.html页面
2. 用户填写用户名，密码，验证码
3. 前端js验证输入有效性
4. 单击“登录”按钮，将请求提交到 `/webapi/v1/UserInfo/login`
5. 后台登陆验证流程

6. 后台验证成功，跳转到main.html
   后台验证失败，提示失败信息

验证码获取

验证码图片是后台的一个请求，请求地址：

```
<img src="/webapi/v1/Util/ValidateCode?s=login&_d=1569393561350">
```

> 参数s是验证码类型；
> 参数_d是当前时间戳，用于保证不使用缓存请求结果

后端收到验证码请求，会生成一个token，并将该token保存到名称为“DMSPNetHMac_”+s(本例中为DMSPNetHMac_login)的cookie中以供前端使用，同时生成一个验证码并将该验证码生成图片，将该图片的二级制数据写进请求的响应结果中
前端收到的响应并展示在页面上。

整体登陆验证流程

1. 用户单击“登录”按钮；
2. 前端js获取cookie中DMSPNetHMac_login的token；
3. js将密码加密；
4. js获取当前时间戳，并将时间戳和token一起加密为hash；
5. 将表单元素值和计算结果组成以下结构：

```
{
    "LoginName":"admin",
    "Pwd":"06ee915a81fb1707fe44a02b83d340be","ValidCode":"ffff","Token":"597b03ad8e9e4d06a8
    "Version":"3","Hash":"61d73a2df335f2c27496583af06222bc","RememberMe":false
    }
```

以上各值意义如下：

字段名称	说明
LoginName	用户名
Pwd	加密后的密码
ValidCode	验证码
Token	token， `cookie['DMSPNetHMac_login']` 的值
Tick	当前时间戳

字段名称	说明
Version	版本号
Hash	hash
RememberMe	是否记住登录状态，记住后一周内免登陆，否则20分钟后登录失效

6. 将以上json数据提交到 `/webapi/v1/UserInfo/login`
7. 后台收到登录请求，验证请求参数，以下任一验证失败均返回登录失败
   a. 验证当前用户或IP是否被锁定
   b. 验证用户名和密码是否为有效字符串
   c. 验证token是否有效
   d. 验证验证码是否正确
   e. 验证用户标识 `userAgent` 是否和请求验证码时的用户标识 `userAgent` 一致
   f. 验证hash是否有效
   g. 验证用户名和密码是否对应数据库的用户记录
   h. 验证用户当时被冻结
8. 若验证失败，记录失败信息，并执行锁定策略
9. 若全部验证通过，登录成功，
   a. 将登录记录到表 `LoginRecords` ，
   b. 将登录记录加密作为会话标识放入响应的头信息中
   c. 返回用户信息和菜单信息,
   d. 前端从请求响应中获取会话标识

登录流程其他说明

1. 若未启用单账号多点登录，每个用户只有最后一次登录有效，之前的登录会提示用户在别处登录，并强制失效
2. 登录成功后会强制中断当前用户的所有找回密码流程

其他请求验证流程

> 只有设置了 `IAuthAttribute` 过滤器的请求会进行以下验证

1. 前端将登录时获取的会话标识放入请求头 `LoginUserToken` 中
2. 发送请求
3. 后端从请求头中获取会话标识 `LoginUserToken` ， 开始验证流程
   a. 验证会话标识信息
   b. 解密会话标识信息，根据会话标识信息获取登录记录，
   c. 验证会话标识和登录记录

d. 验证登录是否已经失效
e. 验证当前用户是否已经在别处登录
4. 若任一验证失败，中止请求
5. 若验证成功，更新会话失效时间，执行后续业务请求

登录记录表LoginRecords

字段名	类型	说明
ID	VARCHAR(36)	主键
UserID	VARCHAR](36)	用户ID
UserName	VARCHAR(64)	用户名
LoginToken	VARCHAR(128)	登录标识
SessionToken	VARCHAR(128)	会话标识
UserAgent	VARCHAR(256)	用户请求标识
ClientIP	VARCHAR(32)	客户端IP地址
ClientHostName	VARCHAR(64)	客户端主机名称
LoginTime	DATETIME	登录时间
ExpiryTime	DATETIME	失效时间
IsRemembered	BIT	是否记住登录状态
IsMandatoryExpiration	BIT	是否已经失效
MandatoryExpirationTime	DATETIME	强制失效时间
CreateTime	DATETIME	记录创建时间

找回密码流程

1. 用户在登录页面点击“找回密码”按钮
2. 跳转到FindPassword.html页面
3. 用户填写用户名和验证码，单击“确定”按钮
4. 后端检查验证码是否有效，验证用户名是否存在，验证失败中断流程
5. 用户填写用户预留的邮箱地址
6. 后端验证用户填写的邮箱是否与预留的邮箱一致，如果不一致，中断流程

7. 生成一条找回密码的记录保存到表 `FindPwdRecords` ，
8. 发送一个定时失效的重设密码的连接到用户邮箱，
9. 用户从邮箱访问重设密码的连接，
10. 后端验证该连接的有效性，若连接无效，中断流程
11. 连接有效，则展示重设密码的界面
12. 用户两次填写一致的密码，并单击“确定”按钮
13. 重设密码成功，强制使当前的找回密码记录失效

找回密码记录FindPwdRecords

字段名	类型	描述
ID	VARCHAR(36)	主键
UserID	VARCHAR(36)	用户ID
UserName	VARCHAR(64)	用户名
Token	VARCHAR(128)	找回密码的凭证 通过随机算法生成的无重复字串
Email	VARCHAR(128)	邮箱
CreateTime	DATETIME	发起时间
ExpiryTime	DATETIME	失效时间
IsExpiried	BIT	状态 0正常，1已失效

其他说明

系统会在每次用户登录时清除一部分历史数据，清除内容包括：

1. 70天以前的登录记录，
2. 70天以前的审计日志
3. 锁定策略最长时间段以前的登录错误记录
4. 锁定策略最长时间段以前的用户锁定记录

> 因为每次登录都有这个操作，所以每次操作都不会有太多需要删除的数据，对效率的影响相对于网络传输速度的影响来说可以忽略不记