

目录

目录	1
Home	3
TMS团队协作	3
核心功能	3
团队沟通	3
团队沟通的子功能	3
团队博文(wiki)	3
团队博文的子功能	4
国际化翻译i18n	4
敏捷任务缺陷跟踪管理	4
TMS	5
TMS(Teamwork Management System)	5
GIF show	5
具有以下功能:	7
沟通功能	7
团队博文(wiki)	7
国际化翻译管理	8
其他功能	8
许可(license)	8
部署指导	9
TMS安装部署 (docker-compose)	9
依赖服务	9
依赖镜像	9
打tomcat war包	9
tomcat镜像制作	9
mysql镜像制作	9
docker-compose启动服务	9
服务访问	10
TMS-md2pdf服务模块部署安装 (博文导出PDF)	11
配置系统可执行node的安装路径	11
默认安装目录	11
可能遇到字体问题	11
TMS安装部署 (传统方式)	12
源码仓库	12
从源码部署后端服务	12
从源码部署前端服务	12
直接部署打好的部署war包(推荐此种方式, 源码编译安装需要编译环境,比较费事)	13
前提需要	13
mysql准备	13
tomcat war包部署	13
配置文件修改	13
Markdown to pdf服务配置	14
注意事项	14
TMS静态资源nginx承载从而加速访问	15
开发指导	17
TMS后端开发技术拆解	17

使用核心技术	17
TMS前端使用技术拆解	18
使用技术	18
运行打包	18
安装nodejs工程依赖	18
开发模式	18
生产构建	19
使用三方插件	19
如何在开发工具中运行	20
开发工具	20
下载源码	20
导入开发工具	20
配置数据库 (推荐mysql5.6)	20
启动	20
Done! Enjoy!	20
TMS开发环境	21
开发工具	21
配置指导	22
通过jenkins插件与jenkins打通	22
具体配置方式:	22
通过webhooks与JIRA打通	24
具体配置方式:	24
集成kkFileView实现常见文件的在线预览	25
kkFileView部署 (单独部署即可)	25
TMS配置	25
常见问题	26
博文word导入时保存失败 (post body过大)	26
windows下前端依赖包问题	27
TMS参与贡献	28
下面列举一些参与方式, 但不限于这些方式.	28
TMS代言~	29
TMS QQ交流群	30
TMS开源团队协作交流QQ群	30
TMS作者	31
关于他	31
他造过的轮子	31
他现在干啥呢?	31
个人网站(他很扣, 不爱写字码文)	31
联系他	31
TMS适合怎样的团队	32
扁平架构的团队组织	32
缓和的通知机制	32
透明	32
分享讨论学习进步	32
倡导追求极简	32
TMS开发蓝图	33

Home

TMS团队协作

TMS 系统诞生的初衷是: 为了解决项目团队协作效率不畅问题, 作者本人就经历过沟通协作不畅的开发团队, 团队使用QQ, 微信等这种实时通讯工具来用于工作办公沟通, 这类工具的实时性是它的长处, 对于一些团队也是一种弊端, 它的实时性首先会干扰到团队成员的集中精力的工作状态, 打断思考解决问题的过程, 其次, 它的实时性会让团队成员在回答问题, 解决问题时缺少思考的过程, 导致沟通就像面对面的对话, 无暇思考, 便急于应答, 会导致沟通者心情不悦, 导致需要协作的问题不能很好解决, 更多的问题是需要思考调研讨论才能找到好的解决方案。

核心功能

团队沟通

TMS沟通相比较QQ一类的实时沟通工具对使用的干扰缓和了许多, 它利用传统邮件办公的模式, 对于有目标性的消息, 可以通过@到指定人, 从而将消息以邮件的形式发送给到指定人, 而指定人只需要根据自己的工作时间安排, 定期查看邮件, 就不会错过重要的消息. 团队沟通的消息也可以集中起来, 以便随时查找使用(这些沟通的消息对于公司和团队也是一种重要的资产沉淀, 价值不菲).

团队沟通的子功能

- 私聊
一些敏感或者涉及安全的沟通消息可以以私聊的方式和沟通者直接沟通.
- 频道(群聊)
团队成成员可以根据需要组建特定成员的沟通频道, 以便满足需要更多成员沟通协作的工作场景, 已达到信息互通共享.
- 频道话题消息(父子消息)
频道成员可以针对某一特定频道消息, 进行话题讨论沟通, 使得特定问题, 集中讨论, 而不至于导致频道消息凌乱, 不易阅读.
- Markdown输入支持
markdown是当前流行的一种简单实用的文本排版组织的语法, 相比较传统的无格式化输入, 可以让沟通者更好的组织自己的思考沟通的内容, 同时也方便其他人的阅读理解.
- 目录(结构化消息目录)
对于包含段落标题的结构化消息内容, 自动生成标题目录结构, 通过点击目录标题条目, 快速定位到对应标题内容区域, 方便阅读者.
- @消息集中查看
集中管理来自他人的@消息, 重要消息不会错过, 在自己规划的时间可以集中查看处理应答, 提供便捷贴心的一些常用操作.
- 收藏消息集中查看
集中管理自己因为重要或者可以稍后处理而收藏的消息, 在自己规划的时间可以集中查看处理应答, 提供便捷贴心的一些常用操作.
- 附件集中查看
集中的附件管理, 包括图片和非图片文件的管理. 图片可以幻灯片式的舒心预览, 附件方便的过滤查找或者下载.
- 团队日程规划
团队协作, 离不开会议, 通过日程规划, 确定时间, 地点, 参与人员发起会议, 会议会邮件通知到参会人员, 在会议开始前web也会提醒参会人员. 从而避免措手不及的临时安排, 随叫随到的会议, 干扰工作.
- 待办事项 (新功能)
工作过程中会累积不少近期要处理的待办事项, 事项如果一多, 就容易导致, 记忆混乱, 进而导致处理效率降低, 通过记录待办事项, 标记其状态, 排序处理优先级, 让事项有条不紊的被消化处理掉.
- 频道消息固定消息
频道成员可以将认为重要的消息, 固定起来, 以便成员随时找到查看.
- 沟通消息标记表情&标签
通过标记, 让消息具备状态标识, 更易区别, 表情投票, 让沟通更具乐趣, 互动起来. 也节省打字的时间; 同时支持按标签快速过滤和查找.
- 沟通消息检索
沉淀的消息在以后的某个时候如果被需要, 这时可以通过检索功能找到, 这些宝贵的资产可不能像QQ, 微信消息一样被埋没, 都是团队的心血资产.

团队博文(wiki)

一个团队如果没有一个可以分享知识经验的地方, 好比没有下蛋场所的母鸡, 团队成员的经验积累不能孵化成形, 造福他人; 通过一个团队博文系统团队成员可以分享与收获, 对于团队和负责的产品的的好处不言而喻.

团队博客的子功能

- 博文空间
便于组织博文, 博文权限的隔离控制.
- 博文空间分类目录 按条理归类空间下的博文, 便于后续查阅.
- 博文写作&编辑
流行markdown书写语法, 简单轻松排版, 剪贴板直接粘贴图片, 很便捷省事, 当然拖拽上传, 选择上传也没丢弃; 同时支持排版功能更强大更丰富的传统html富文本编辑器.
- 博文评论, 赞
众人拾柴火焰高, 好的分享博文, 免不了他人的评论, 讨论的过程更是收获的过程. 能博得他人一赞, 也是对作者的勉励.
- 博文历史
博文版本管理, 在某些时候是蛮重要的, 尤其在可以多人协作编辑的博文(tms博文是支持协作的).
- 博文协作
当需要多人协作完成一篇博文时, 这不就是利器嘛.
- 博文收藏,移动,复制, 关注
常用小功能.
- 博文限制
上面提到了空间级别的权限控制, 单独的博文当然也是可以进行权限控制的(指定人, 指定频道可见).
- 博文导出
当博文需要分享给系统外人时, 你可以选择将博文导出为 pdf 或者 markdown 文件, 从而分享给他人.
- 博文分享
可以便捷的将博文分享给指定用户, 频道, 或者直接邮件分享.

国际化翻译i18n

方便对国际化翻译的协作管理,从而让翻译者, 开发者, 管理者, 关注者等多种角色的协作参与.

敏捷任务缺陷跟踪管理

开发中, 敬请期待...

TMS

TMS(Teamwork Management System)

TMS是开源团队协作(沟通,博文,国际化翻译)web系统(响应式界面设计,移动端适配).

repo:
<http://git.oschina.net/xiweicheng/tms>
<https://github.com/xiweicheng/tms>

GIF show

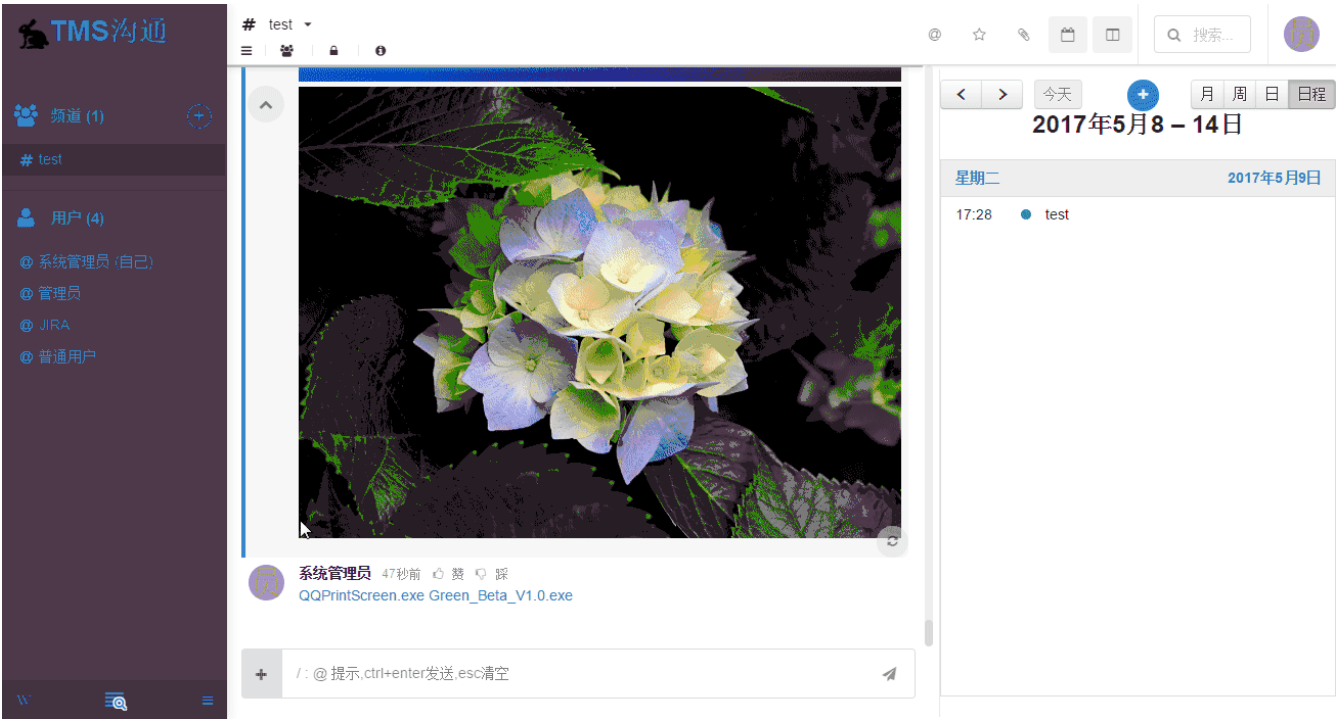
着陆页

国际化翻译



tms000.sh1.newtouch.com/admin/project

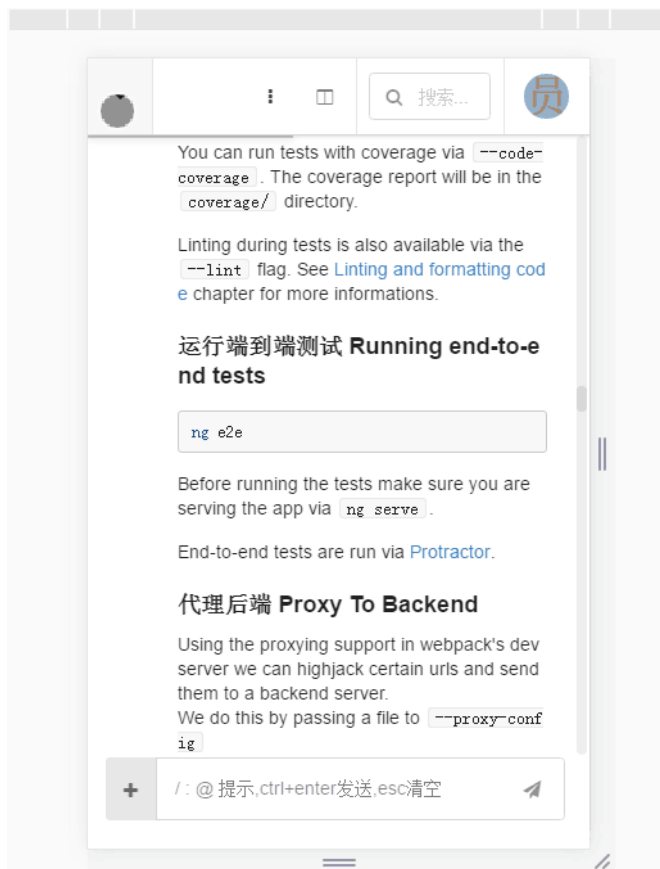
团队沟通



团队博文(wiki)



移动端响应式设计



具有以下功能:

- 团队协作沟通功能(类似于slack bearychat)
- 团队博文(wiki) 类似 精简版confluence 蚂蚁笔记
- 国际化翻译管理.

沟通功能

- 频道(组团沟通)
- 私聊(一对一)
- markdown语法支持(内容排版不再单调)
- @消息 收藏消息 富文本消息目录
- 日程安排(提醒)
- 待办事项
- 剪贴板上传图片, 拖拽文件上传(就是这么便捷)
- 邮件通知 桌面通知(不用担心错过什么)
- 热键支持(鼠标点多了也累不是)
- 更多贴心便捷操作等你发现

团队博文(wiki)

- 博文空间(便于博文组织,权限隔离)
- Markdown书写
- 博文目录
- 博文关注,收藏,历史,权限
- 博文评论
- 导出 pdf & markdown
- 更多贴心便捷操作等你发现

国际化翻译管理

包括以下核心功能模块

- 翻译项目管理
- 翻译语言管理
- 翻译导入导出
- 翻译管理

其他功能

- 系统设置
- 用户管理

许可(license)

MIT

部署指导

TMS安装部署（docker-compose）

注意：下面docker方式部署因为数据没有映射到外部宿主机上或者其它存储外设上，重新构建应用或者DB镜像后部署，数据会丢失（因此仅适合尝鲜或者演示场景）。若要生产部署请选择**传统部署方式**，或者自行优化调整docker部署参数以保证数据安全。

依赖服务

- docker
- docker-compose
- maven
- jdk8

依赖镜像

```
docker pull mysql:5.6
docker pull tomcat:8.5
```

打tomcat war包

1. 下载源码（下载地址：<https://gitee.com/xiweicheng/tms>）
2. 修改 `src/main/resources/application.properties`

```
spring.profiles.active=prod
```

3. 修改 `src/main/resources/application-prod.properties`

注意使用下面配置，不要做任何调整，否则会报错起不来

```
spring.datasource.url=jdbc:mysql://db:3306/tms?useUnicode=true&characterEncoding=UTF-8
spring.datasource.username=root
spring.datasource.password=pingan
```

4. maven打包（注意选择jdk1.8）

tms项目根目录下执行：

```
mvn clean package -Dmaven.test.skip=true
```

tomcat镜像制作

tms项目根目录下执行：

```
docker build -t xiwc/tms .
```

mysql镜像制作

cd到db目录下执行：

```
docker build -t xiwc/tms-mysql .
```

docker-compose启动服务

```
docker-compose up -d
```

服务访问

访问：<http://localhost:8090/#/home>

登录密码：`super/88888888`

注意：如果是云主机或者其他物理机，注意端口是否开放，不然访问不了。

Done, enjoy!

TMS-md2pdf服务模块部署安装（博文导出PDF）

配置系统可执行node的安装路径

修改配置文件：application-tms.properties，下面配置成系统中node的具体路径

```
#可执行bin node目录  
tms.bin.node.path=/usr/local/bin/node
```

默认安装目录

```
/home/tms/md2pdf
```

- 复制 tms/src/main/resources/md2pdf 中全部文件到 /home/tms/md2pdf 中.
- 执行 npm install 或者如果安装了cnpm, 执行 cnpm install
- done.

也可以修改配置指定 md2pdf 服务目录.

配置位置: tms/src/main/resources/application-prod.properties 调整配置: tms.blog.md2pdf.path=/home/tms/md2pdf

可能遇到字体问题

解决方案:

问题: 用phantomjs去截取中文页面的网站可能会出现乱码的情况，也就是截图中中文的位置全是方框。

解决办法就是安装字体

在centos中执行：yum install bitmap-fonts bitmap-fonts-cjk

在ubuntu中执行：sudo apt-get install xfonts-wqy

这样再去截图中文的页面就不会出现一堆的方框了。

TMS安装部署（传统方式）

源码仓库

服务后端: <https://gitee.com/xiweicheng/tms.git>

沟通&博文前端: <https://gitee.com/xiweicheng/tms-frontend.git>

着陆页前端: <https://gitee.com/xiweicheng/tms-landing.git>

从源码部署后端服务

前提需要:

- git
- maven
- jdk8
- tomcat8
- mysql5.6

从源码部署前端服务

前提需要:

- git
- nodejs
- aurelia-cli
- nginx

直接部署打好的部署war包(推荐此种方式, 源码编译安装需要编译环境,比较费事)

前提需要

- jdk8
- tomcat8
- mysql(推荐5.6,其他版本未很好测试过, 5.7存在已知问题,暂请勿用)

mysql准备

创建数据库

```
CREATE DATABASE ${db_name} DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
```

`${db_name}` 替换为想设定的数据库名称.

可选配置: 如果 沟通消息&博文 要支持宽字节字符(例如emoji表情字符), 需要手动修改表结构的 `content` 字段为 `utf8mb4` 编码. 一般修改四个表 (表会在web服务初次启动完成后通实体映射自动生成)就应该可以了:

名	自动递增	修改日期	数据长度	表类型	行	注释
language	3		16 KB	InnoDB	2	
label	3184		272 KB	InnoDB	3102	
groups	12		16 KB	InnoDB	9	
group_members	389		16 KB	InnoDB	75	
group_authorities			16 KB	InnoDB	0	
file_translate			16 KB	InnoDB	3	
file	2166		416 KB	InnoDB	2154	
feedback	163		64 KB	InnoDB	159	
comment	463		336 KB	InnoDB	451	
client_details			16 KB	InnoDB	0	
chat_stow	192		16 KB	InnoDB	74	
chat_reply	91		16 KB	InnoDB	79	
chat_pin	25		16 KB	InnoDB	7	
chat_label	129		16 KB	InnoDB	93	
chat_direct	1101		1552 KB	InnoDB	871	
chat_channel	1691		1552 KB	InnoDB	1562	
chat_at	6203		1552 KB	InnoDB	6168	
chat	1538		1552 KB	InnoDB	1390	
channel	36		16 KB	InnoDB	15	
blog_stow	30		16 KB	InnoDB	27	
blog_history	293		1552 KB	InnoDB	285	
blog_follower	18		16 KB	InnoDB	17	
blog_authority	2		16 KB	InnoDB	0	
blog	199		1552 KB	InnoDB	183	
authorities			16 KB	InnoDB	114	
actor_schedule			16 KB	InnoDB	55	

tomcat war包部署

这里下载war包(新版本不再提供war包,需要自行maven打包): <http://git.oschina.net/xiweicheng/tms/releases>

补充: **TMS后端代码**已经包含了前端的build资源, 所以前端代码无需打包构建. 直接基于后端代码进行maven打包即可. 打包命令 `mvn clean package -Dmaven.test.skip=true`

将war包解压到tomcat的`**webapps/ROOT/**`下面(解压前清空ROOT目录下内容).

ps: war解压命令 `jar -xvf tms-xxx.war`, 或者通过其他解压缩工具.

配置文件修改

配置位置: `webapps\ROOT\WEB-INF\classes`

- application.properties修改 `spring.profiles.active=dev` 为 `spring.profiles.active=prod`, 从而使用生产环境配置.
- application-prod.properties修改

数据库连接信息配置(替换下面配置中的`{}`为实际配置信息)

```
spring.datasource.url=jdbc:mysql://${ip}:${port}/${db_name}?useUnicode=true&characterEncoding=UTF-8
```

```
spring.datasource.username=${db_username}
spring.datasource.password=${db_password}
```

Markdown to pdf服务配置

该服务是在将博文 导出为pdf 需要, 不配置该服务, 不会影响服务的运行, 只是 导出为pdf 无法正常使用.

FYI: [tms md2pdf服务模块部署安装](#)

注意事项

- 系统附件 上传的附件信息在 `webapps\ROOT\upload\` 目录中, 请注意不要误删除. 如果需要备份, 直接备份整个该upload目录即可.
- 数据库备份 请自行拟定备份方案
- 内置管理用户

super/88888888 系统管理员, 拥有全部系统权限.

- 博文下载pdf 该功能需要服务端配置**md2pdf**服务支持(是一个nodejs模块), 调用路径配置在 `application-prod.properties` 中, 为:
`tms.blog.md2pdf.path=/home/tms/md2pdf`, 可以根据模块位置自行调整配置.

TMS静态资源nginx承载从而加速访问

nginx开启gzip压缩,再将动态api数据请求通过nginx反向代理转发到tomcat.

配置示例:

```
#user nobody;
worker_processes 4;

error_log logs/error.log;

pid logs/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include mime.types;
    default_type application/octet-stream;

    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log logs/access.log main;

    sendfile on;
    tcp_nopush on;
    tcp_nodelay on;
    keepalive_timeout 65;
    types_hash_max_size 2048;

    gzip on;
    gzip_http_version 1.0;
    gzip_min_length 1024;
    gzip_buffers 4 16k;
    gzip_comp_level 3;
    gzip_proxied any;
    gzip_disable "MSIE [1-6].";
    gzip_types text/plain application/json application/x-javascript text/css text/javascript application/octet-stream application/javascript application/xml application/

server {
    listen 9090;
    server_name 10.47.58.89:9090;

    root /data/apps/tomcat/webapps/ROOT/WEB-INF/classes/static;

    location = / {
        expires -1;
        rewrite / /index.html break;
    }

    location ~ ^/(lib)/ {
        expires 30d;
    }

    location = /page/index.html {
        expires -1;
    }

    location = /page/blog.html {
        expires -1;
    }

    location = /page/gantt/index.html {
        expires -1;
    }
}
```

```
location ~ ^/(admin/(js|css|img|md)|img|landing|page|scripts|themes|swipebox)/ {
    expires 12h;
}

location / {

    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header Host $host;
    proxy_http_version 1.1;

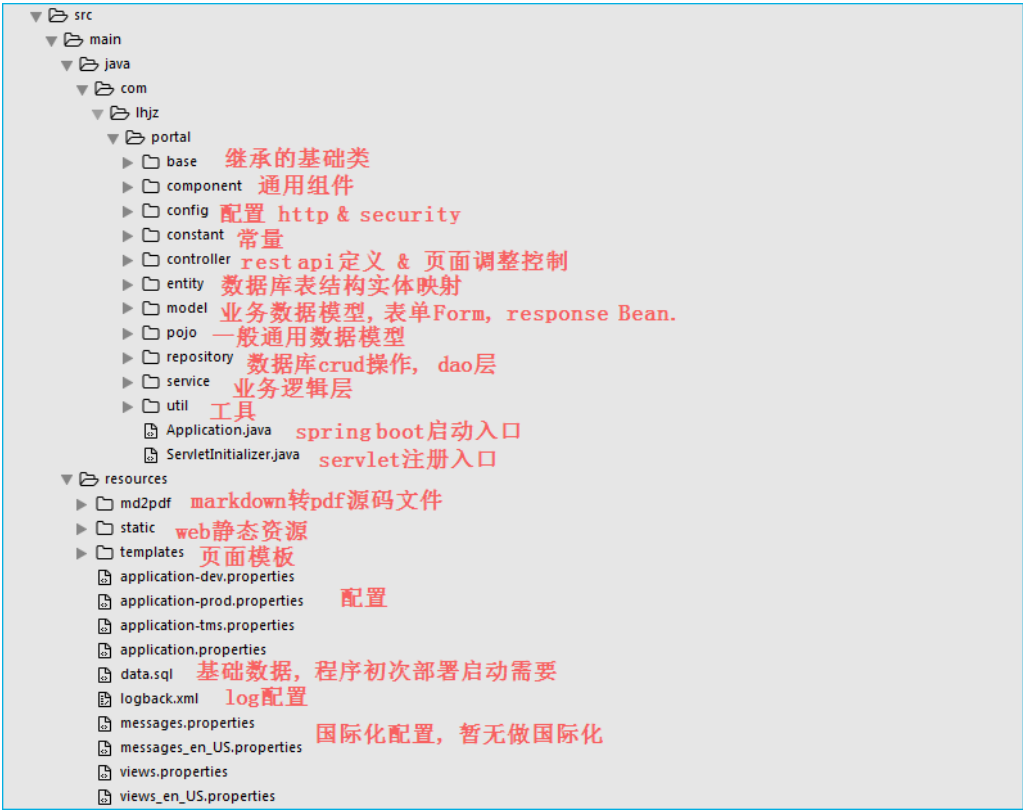
    proxy_headers_hash_max_size 51200;
    proxy_headers_hash_bucket_size 6400;

    #proxy_redirect off;
    proxy_set_header Host $host:$server_port;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header REMOTE-HOST $remote_addr;
    proxy_set_header X-Forwarded-Host $host;
    proxy_set_header X-Forwarded-Server $host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    client_max_body_size 50m;
    proxy_pass http://localhost:8080/;
}
}
```


开发指导

TMS后端开发技术拆解

为了便于有想法贡献该项目的人参与协作开发, 或者想基于该项目进行二次开发.



使用核心技术

<http://projects.spring.io/spring-boot/>

<http://projects.spring.io/spring-security/>

<http://projects.spring.io/spring-data-jpa/>

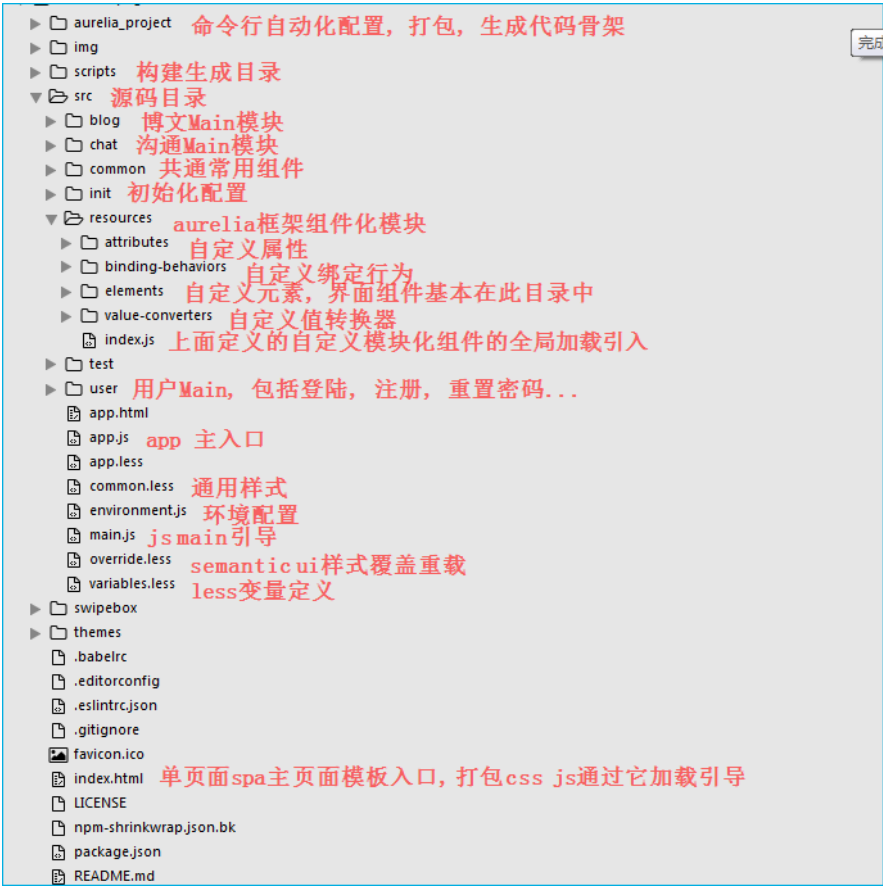
<http://www.thymeleaf.org/>

- 通过maven管理项目
- 使用mysql数据库(推荐5.6版本, 5.7存在已知问题, 需要修改默认配置)

TMS前端使用技术拆解

为了便于有想法贡献该项目的人参与协作开发, 或者想基于该项目进行二次开发.

前端项目: <http://gitee.com/xiweicheng/tms-frontend>



使用技术

- UI: semantic-ui 一款精美的语义化前端设计语言, 组件丰富.
- 框架: aurelia 一款双向绑定JavaScript客户端框架.

项目基于 aurelia-cli: <http://aurelia.io/hub.html#/doc/article/aurelia/framework/latest/the-aurelia-cli/1>

运行打包

安装nodejs

全局安装 aurelia-cli: `npm install aurelia-cli -g`

安装nodejs工程依赖

mac下直接下载 https://gitee.com/xiweicheng/tms-resources/blob/master/tms-frontend_node_modules.zip 解压到更目录下即可。

windows下参考贡献者提供的 (压缩包内包含指导文档) : https://gitee.com/xiweicheng/tms-resources/blob/master/windows_tms-frontend.zip

`npm install`

开发模式

```
au run --watch
```

生产构建

```
au run build --env prod
```

使用三方插件

<https://github.com/flesler/jquery.scrollTo>

<https://github.com/chjj/marked>

<https://www.npmjs.com/package/highlight.js>

<https://github.com/zenorocha/clipboard.js>

<https://github.com/lgarron/clipboard.js>

<https://www.npmjs.com/package/dropzone> <https://github.com/enyo/dropzone>

<https://github.com/ayerssss/paste.js>

<https://github.com/hustcc/timeago.js>

<https://www.npmjs.com/package/swipebox>

<http://www.jacklmoore.com/autosize/>

<https://www.npmjs.com/package/jquery.hotkeys>

<https://github.com/jeresig/jquery.hotkeys>

<https://github.com/yuku-t/jquery-textcomplete>

https://github.com/yuku-t/jquery-textcomplete/blob/master/doc/how_to_use.md

<https://www.npmjs.com/package/aurelia-textcomplete>

<https://github.com/NextStepWebs/simplemde-markdown-editor/>

<http://gromo.github.io/jquery.scrollbar/demo/basic.html>

<https://github.com/gromo/jquery.scrollbar>

<https://github.com/zenozeng/color-hash>

<https://github.com/Nickersoft/push.js>

<https://github.com/PitPik/tinyColorPicker>

<http://fancyapps.com/fancybox/3/>

如何在开发工具中运行

开发工具

下载开发工具sts(其实也就是eclipse) : <https://spring.io/tools/sts/>

sts安装lombok插件 : <https://blog.csdn.net/dorothy1224/article/details/79280591/>

下载源码

下载源码zip打包或者git clone项目到本地

导入开发工具

选择maven方式导入下载的项目

配置数据库 (推荐mysql5.6)

1. 创建一个数据库(例如:tms)

```
CREATE DATABASE tms DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
```

2. 配置数据库(application-dev.properties)

```
spring.datasource.url=jdbc:mysql://localhost:3306/tms?useUnicode=true&characterEncoding=UTF-8
spring.datasource.username=xxx
spring.datasource.password=xxxxxx
```

启动

直接运行 Application.java 的main方法即可

页面访问 : <http://localhost:8080/> 默认账号 : super / 88888888

Done! Enjoy!

TMS开发环境

开发工具

后端: [spring sts\(eclipse\)](#) + maven + [lombok\(插件\)](#) + [jdk8](#)

涉及技术:

- [spring boot](#)
- [spring security](#)
- [spring data jpa](#)
- [thymeleaf](#)

前端: [sublime text 3](#) ([vscode](#)) + [nodejs](#) + [aurelia-cli](#)

涉及技术:

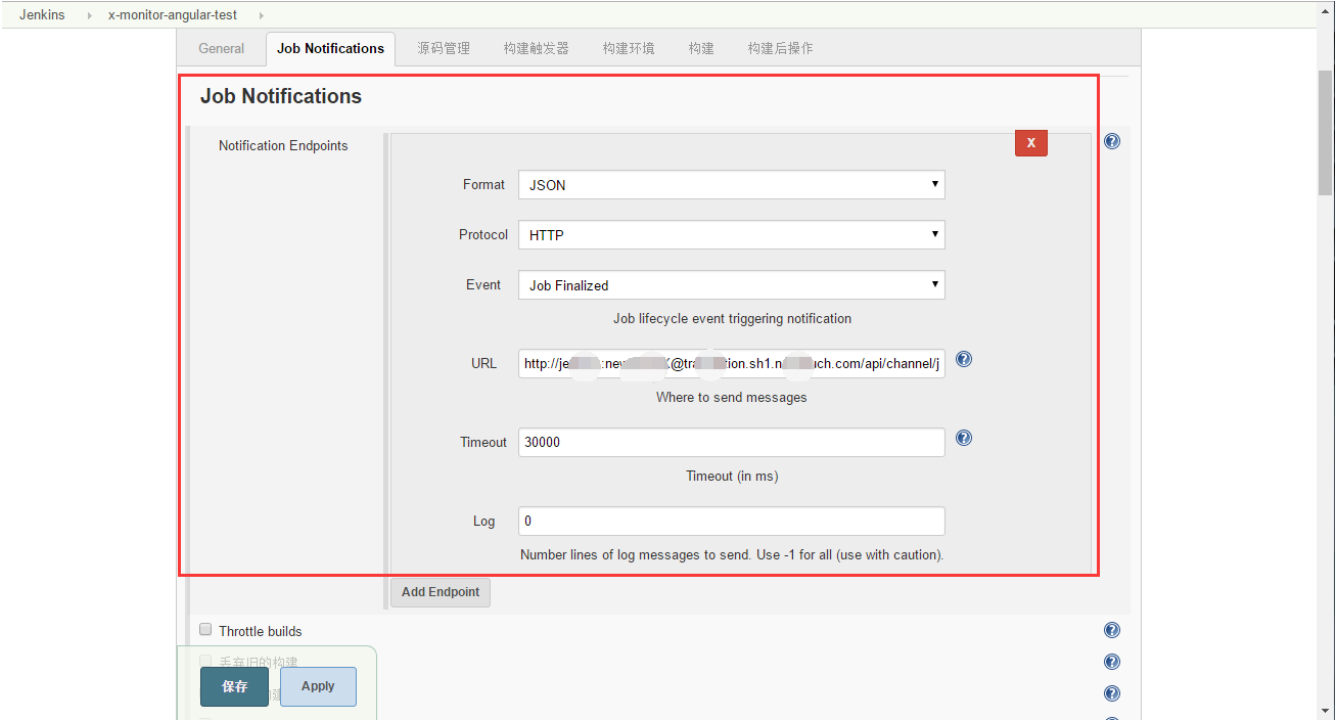
- [aurelia](#)
- [semantic-ui](#)

配置指导

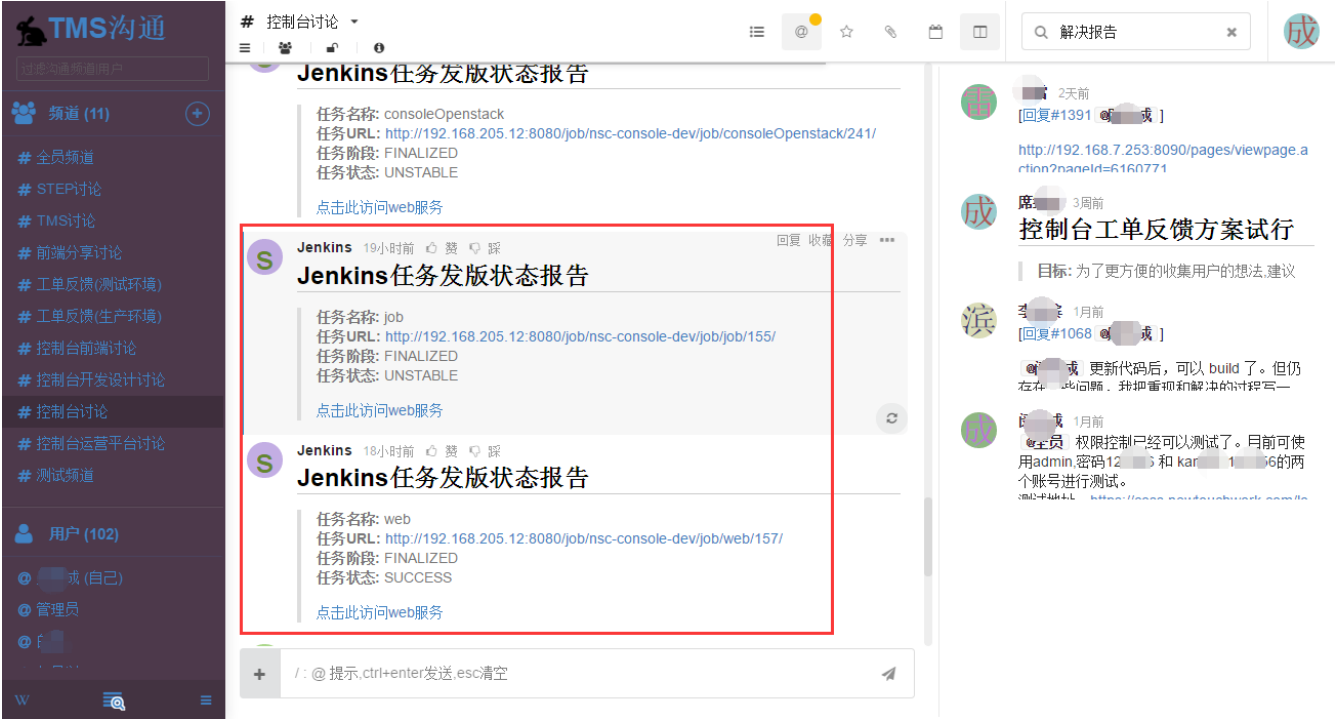
通过jenkins插件与jenkins打通

通过jenkins的 Job Notifications 插件可以和tms集成, 从而通过tms特定频道接收jenkins任务最终状态结果推送.

jenkins配置:



TMS集成效果截图:



具体配置方式:

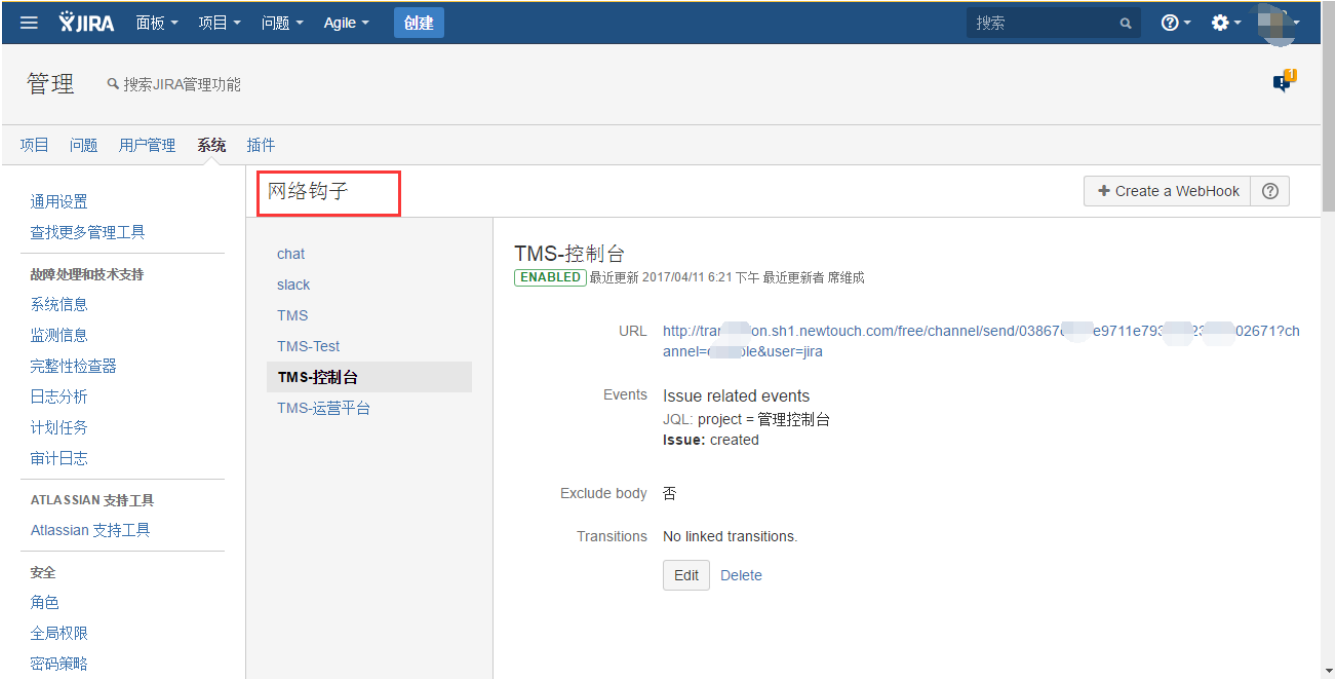
```
http://username:password@192.168.1.1:8080/api/channel/jenkins/send?channel=test&raw=true&web=http://xxx.com
```

TODO...

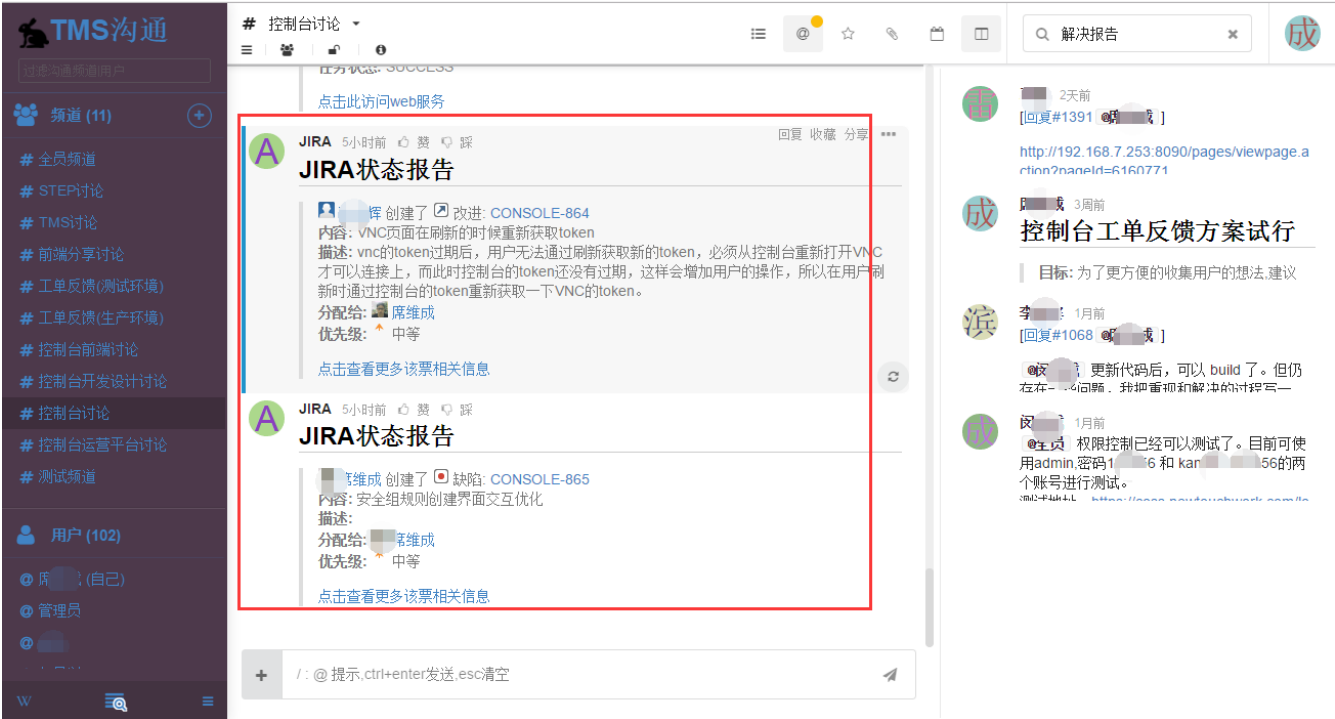
通过webhooks与JIRA打通

通过jira的网络钩子功能和TMS集成打通, 让TMS特定频道接收JIRA的问题创建更新等事件通知.

JIRA配置:



打通截图演示:



具体配置方式:

TODD...

集成kkFileView实现常见文件的在线预览

kkFileView部署（单独部署即可）

参考：<https://kkfileview.keking.cn/zh-cn/docs/production.html>

TMS配置

修改配置路径：`src/main/resources/application-prod.properties`

注意：开发环境配置到 `src/main/resources/application-dev.properties`

修改下面配置，属性值为kkFileView服务的地址（例如：<http://127.0.0.1:8012>）

```
tms.file.online.view.url=
```

配置完成，服务重启即可。

常见问题

博文word导入时保存失败（post body过大）

因为tomcat默认支持的最大post请求body是2M，需要按需扩大最大限制。

可以设置 `server.xml` 中 `connector` 标签的属性 `maxPostSize`

tomcat：默认大小 2097152 （2M），例如可以修改设置 `maxPostSize="20971520"` 时，设置为20M

参考：

- tomcat8 `maxPostSize="0"` 不生效: <https://blog.csdn.net/u013350170/article/details/54963631>
- 关于tomcat不同版本的`maxPostSize`配置差异: <https://www.jianshu.com/p/2d7b489940ef>

windows下前端依赖包问题

参考贡献者整理的解决方案（压缩包内含指导文档）：

- https://gitee.com/xiweicheng/tms-resources/blob/master/windows_tms-frontend.zip

TMS参与贡献

下面列举一些参与方式, 但不限于这些方式.

- 通过 **Issues** 提交使用过程中发现的UI体验问题, 系统bug.
- 提出觉得有价值的功能需求, 用户使用场景.
- 参与到该项目的讨论中来. [点击链接加入群【TMS开源团队协作交流】](#)
- 为项目贡献代码, 成为tms协作开发者, 共同推动一款好的团队工具诞生流行
- 分享tms给有类似需求的周边朋友, 团体或者参与的社区
- 企业小团队部署定制化开发的有偿服务(为开源作者的付出挣点外快).
- 赞助该项目(十块八块不嫌多, 一毛两毛不嫌少)

如何赞助



- 或者通过项目提供的赞助入口赞助.

TMS代言~

如果您使用了TMS, 欢迎留下自己的公司名, 或者商标, 官网地址, 以便TMS将您的品牌加入到TMS的官网首页中,做品牌推广...
如果您不想自己的系统地址留在下面, 也请留言告知, 这边给予删除.

作者邮件: xiwc87@yeah.net

- <http://tms000.sh1.newtouch.com/>
- <https://xiweicheng.com/>
- <http://tms.yhskyc.com/>
- <http://wiki.fuxiangtong.net/>

TMS QQ交流群

TMS开源团队协作交流QQ群

群号: 274832399

[点击链接加入群【TMS开源团队协作交流】](#)



TMS作者

关于他

- 他喜欢好的工具, 因为地球人的进步就是基于工具的不断发明与革新.
- 他对于低效的工作模式, 有改进的意愿与想法, 让大家都省心省事完成工作, 按时下班, 放松身心健康体魄.
- 他在自己的工作历程中, 基于自身或者团队需求设计开发了很多的工具, 解放了自己与团队, 现在他想通过解放自己与团队的工具去解放更多的人.

他造过的轮子

- P2P模式的Java聊天软件 <http://down.51cto.com/data/257638>
- 类似QQ的绿色截图软件 <https://gitee.com/xiweicheng/QQPrintScreen>
- 基于微信的点餐平台(N年前的设计想法,如今普及了,哎...没坚持去推广) <https://github.com/xiwc/czs>
- 剪贴环(为经常复制粘贴的人造的) <https://github.com/xiwc/clipboard-ring>
- 系统文件检索收藏助手(工作需要的文件太多, 组织在一块好管理) <http://down.51cto.com/data/257662>
- 聊天QQ语音视频, 文件传输, 桌面共享, 远程控制 <http://down.51cto.com/data/271488>
- 还有有些与项目相关的工具, 慢慢翻找, 看能否找到吧

他现在干啥呢?

他, 当前在上海这个套路深的大都市里售卖自己的青春呢!!!

个人网站(他很扣, 不爱写字码文)

<https://xiweicheng.com/>

联系他

QQ: 547538651

邮箱: xiwc87@yeah.net

TMS适合怎样的团队

扁平架构的团队组织

频道消息, 博文的查看分享权限是对等的, 它没有阶级色彩的权限控制. 大家言论自由, 分享自由!

缓和的通知机制

通过 @特定人 & 分享给特定目标, 通知内容以邮件的形式发送通知到目标人, 目标人根据自己的时间安排定时查收处理回复, 避免工作干扰, 无思考仓促的回答.

透明

没有那些对自己团队成员保密性质的东东.

分享讨论学习进步

积极的互帮互助的, 知识共享, 共同进步的团队.

倡导追求极简

TMS的UI设计, 操作流程去繁就简, 简单清爽, 随你所需.

TMS开发蓝图

欢迎讨论

- ☐ tms-frontend 添加自动化脚本, 实现打包并且将打包的文件复制到 tms, 实现git 提交推送.
- ☐ 频道订阅接收的邮件标题调整, 当前为@消息模板.
- ☐ TMS敏捷|看板任务管理.
- ☐ TMS更新公告展示设计实现(代码结构已经整理优化, 准备...)
- ☐ 沟通消息, 博文评论 支持分词检索
- ☐ 博文检索, 上下方向键选择, 选中项目滚动至可见.
- ☒ 邮件模板改进优化
- ☐ gitosc代码质量检查问题对应处理
- ☐ 沟通消息回复, 检测消息中的相关 @user 添加到回复消息FYI中
- ☐ 代码添加版权注释信息
- ☐ 博文创建保存弹窗添加选项: 游客可见
- ☐ 博文评论 & 沟通消息折叠功能(作者&管理员可折叠)
- ☒ 用户自动加入全员频道 TMS频道
- ☐ 频道公告展示
- ☐ 着陆页添加 产品 服务 联系我们
- ☒ 沟通回复右侧边栏展示, slack的Thread功能
- ☒ 频道消息 Pin功能(固定频道消息, 以便成员查看)
- ☒ 沟通消息表情状态 +1
- ☐ 博文seo
- ☐ 自定义沟通消息回复
- ☐ 博文评论 Emoji&标签功能
- ☐ 申请沟通消息编辑权限.
- ☐ 用户过滤查询功能.
- ☒ 移动端消息输入内容删除不正确问题. (修改simplemde) 遗留问题 >
- ☒ 话题消息回复消息定位查看 eg: <http://translation.sh1.newtouch.com/page/index.html#/chat/test?id=11&rid=1>
- ☒ 话题消息, 可以关注, 从而接收更新后者讨论邮件通知.
- ☒ 图片灯箱插件更换为fancybox3: <http://fancyapps.com/fancybox/3/>
- ☒ 话题消息, 左侧边栏 hover 表示
- ☒ 频道固定消息定位失效
- ☒ 回复消息的收藏,刷新
- ☐ 回复消息的消息标签
- ☐ @消息, 收藏消息, PIN消息, 显示回复消息状态, 标签.
- ☐ 频道消息检索(包括话题消息的回复消息)
- ☒ 话题消息回复自动关注被回复的话题消息
- ☐ (邮件中可以提醒, 取消关注该话题, 以避免邮件干扰)
- ☒ 频道消息提醒设置(@消息 & 新消息)
- ☒ 频道外链汇总设计展现
- ☐ 沟通收藏内容片断(可快速复制到剪贴板)
- ☐ 检索focus最近博文展示推荐
- ☐ 这里考虑去掉:

- ☑ 订阅消息邮件提醒(排除自己发送的消息提醒)
- ☑ 博文左侧系统外链弹出导致UI抖动问题.
- ☑ 邮件发送队列设计,解决有时邮件发送失败问题.
- ☑ 博文, 我的收藏 博文列表没有排序处理.
- ☑ 频道删除时,不解除和频道成员的关系(或者保留频道创建者成员). enhancement
- ☑ tms沟通logo点击跳转到门户首页 enhancement
- ☑ 系统|频道外链,统计点击次数, 最后点击时间, 以便决策是否不常用可以移除 enhancement

- ☐ 订阅关注邮件每个用户单独发送
- ☐ tms登录页面改进
- ☐ 博文批量管理
- ☐ github三方账号登录 置顶 feature
- ☒ 消息不是实时推送的啊 question (不解决)
- ☐ 博文导出为word, word导入功能 feature
- ☐ 博文左侧目录栏可见定位功能 enhancement

TMS开发蓝图

- ☑ TMS门户首页重新设计 feature
- ☑ 博文门户页面设计
- ☑ 博文游客可见功能 置顶 feature
- ☑ 访客可见博文展示UI设计开发 feature

