

集创 OA 系统开发文档

2014 年 10 月 11 日 18:22:09

一、需求分析阶段

1、**项目背景**：主要是没啥需求背景，主要是集创的团队主页和团队内部的管理系统这样的东西，类似于单/多团队成员间的交互系统和管理系统，有点像员工管理系统，同时具备一些功能，代码托管，项目发布，垂直网络搜索互联网上的资源并进行后台下载并存储。有点类似于云服务，有点像云管家的感觉。还有给出优酷的连接分析出 flv 的地址并下载，然后再前端的网页进行 html5 播放器的播放。如果不支持的话最好内嵌转码功能。

2、功能分析：

(1) 团队成员的签到，登录，注册，信息修改，即 Member 的增删改查业务

(2) Member 对象类的文件树 (FileTree)，形成一个用户专用的存储空间用于托管代码，存储文档和文件，进行离线下载等工作，和团队成员呈一对一关系。

(3) 虽然现在集创只有一个项目组 (Team)，但是也要预留接口，即项目组的增删改查操作。和团队成员是多对多关系。

(4) 增加一个采购 (Buy) 业务，同时引入权限对象 (Role)，具有某些权限的成员具有申请采购业务，和团队成员之间多对多

(5) 增加一个职务 (Post) 对象 , 和权限对象多对多对应

(6) 和采购业务相对的一个审批业务流程 (Judge) , 只有某些权限的成员具有审批要求是否通过的权限

(7) 项目组成员完成任务 (Task) , 也需要有审批业务流程 (Judge) , 由该项目组高权限 (Role) 的人确认是否通过 , 以计算任务完成百分比。

(8) 项目组也有对应的目录树 , 这里就有两个选择。一个是将项目组设为 Member 对象 , 然后使用现成的 FileTrees。第二个是生成一个 Team 对象 , 然后生成一个 TeamFileTree 对象单独列出来

(9) 项目组也有对应的权限和职务的设置 , 每个项目组都可能不一样。

(10) 通知系统 (Notification) 对象 , 由具有某些高权限的人对整个团队 , 项目组 , 或者个人进行简短的信息广播

(11) 私信系统 (Message) 对象 , 团队成员之间的内部邮件 , 单独设置模块来通知 , 具有群发功能 ,

(12) 任务系统 (Task) 对象 , 由项目组高权限者分配给低权限者任务 (Task) 对象 , 和团队成员多对一

(13) 博客 , 学习笔记 (Blog) 对象 , 和团队成员多对一 ,

(14) 项目笔记对象 (ProjectText) 对象 , 和项目组多对一 , 由项目负责人编写

(15) 代码版本控制功能 , 抽象出一个版本对象 (Version) , 通过计算 SHA1 来进行文件的版本 , 内部由文件树构成。

(16) 匿名版论坛功能 , 前台用户留言讨论交流等

(17) 日程 (Cale) 对象 , 标记某日该做什么 , 由管理员定义 , 完成增删改

查。

(18) 新闻 (News) 类 , 前台和后台共享信息。或者设置标记

(19) 个人设置功能

(20) 其他未定义

二、模型分析阶段

一. 团队成员对象 Member —— 数据库表 基本字段如下

1. 团队成员主键 mid int 10 自增长
2. 团队成员用户名 musername varchar 60 可能自定义 , 可能使用 (公司) 邮箱
3. 团队成员密码 mpassword varchar 255 需要用到 AES 加密
4. 团队成员姓名 mname varchar 20 汉字
5. 团队成员注册日期 mdate varchar/date 50 可能会使用 date 来进行数据库计算或者使用 String 到 Java 端来进行计算
6. 团队成员照片 mimage varchar 255 照片路径
7. 团队成员其他信息 mqa varchar 255 使用键值对来进行存储
弹性较大 类似于这样 { " 性别 " : " 男 " , " 身份证号码 " : " 220112xxxx " } ;
8. 团队成员状态 mstatus varchar 50 用户的状态 没有使用 int 更多是为了弹性。因为后台就算是判断也是 session 进行枚举比对
9. 团队成员简历 minfo varchar 1000 用户的简历 使用超文本

格式

二. **权限对象 Role** —— 数据库表

1. 权限主键 rid int 10 自增长
2. 权限陈述 rdes varchar 255
3. 权限状态 rstatus varchar 50
4. 父权限 rrid int 10;

三. **职务对象 Post** —— 数据库表

1. 职务主键 pid int 10 自增长
2. 职务描述 pdes varchar 255 自增长
3. 职务状态 pstatus varchar 255 自增长

四. **审批对象 Judge** —— 数据库表

1. 审批主键 jid int 10 自增长
2. 审批内容 jdes varchar 1000 可能会动用 I/O 存储
3. 审批来源 (发送人员)(团队成员主键) jfrom_mid int 10 仅单一
4. 审批目的地 (审批人员)(团队成员主键) jto_mid int 10 仅单一
5. 审批状态 jstatus varchar 50
6. 审批发送日期 jstartdate varchar 100
7. 审批确认日期 jenddate varchar 100

五. **项目组对象 Team** —— 数据库表

1. 项目组主键 tid int 10 自增长

2. 项目组标题 tname varchar 100
3. 项目组封面图片 timage varchar 255
4. 项目组内容 tdes varchar 255
5. 项目组最新版本树 tvid int 10 (使用了版本对象---他会一直回溯)
6. 关于项目组人员管理使用多对多模型
7. 项目组其他信息 tqa varchar 1000 键值对模型
8. 项目组创建日期 tdate varchar 255
9. 项目组其他信息：键值对 tinfo varchar 255
10. 项目组类型信息 ttid int 10

六. 项目类型 TeamType —— 数据库表

1. 类型 id ttid int 10
2. 类型名称 ttname varchar

七. **个人文件树对象** —— 数据库表 ,每条数据都有一个父文件的外键

/ 也可能使用 XML 进行映射 单单只在数据库中记录 XML 的地址

1. 文件树对象主键 fid int 10 自增长
2. 文件树用户外键 fmid int 10 一对一
3. 文件树 xml 路径地址 fpath varchar 255
4. 文件树状态 fstatus varchar 50

如果使用文件树的话

- 3、文件路径
- 4、文件父节点

5、文件创建日期

八. **项目文件树对象** —— 同上

1. 同上 tfid int 10

2. 项目文件树团队主键 tftid int 10 也是一对一

九. **通知对象 Notification** —— 数据库表

1、通知 id nid int 10

2、通知内容 ndes varchar 10

3、通知时间 ndate varchar 255

4、通知发起人 nmid int 10

5、通知其他信息：键值对 ninfo varchar 255

十. **采购对象 Buy** —— 数据库表

1、采购 id bid int 10

2、采购描述 bdes varchar 255

3、采购状态 bstatus varchar 255

4、采购时间 bdate varchar 255

5、采购申请人 breq int 10 对应单一的

6、采购确认人 bsub int 10 对应单一的

7、采购花费人民币 bcash int 100 //是元乘以 100

8、采购其他状态：键值对 binfo varchar 255

十一. **新闻对象 News** —— 数据库表

1. 新闻主键 id newsid int 10

2. 新闻描述 newsdes varchar 255

3. 新闻事件 newsdate varchar 50
4. 新闻状态 newsstatus varchar 50
5. 新闻撰写人 newsmid int 10
6. 新闻其他信息：键值对 newsinfo varchar 255

十二. **私信对象 Message** —— 数据库表/可能会使用 XML 进行映射

1. 私信主键 id mesid int 10
2. 私信发信人 mesfrom_mid int 10 对应单一
3. 私信收信人 改为多对多
4. 私信标题 mestitle varchar 50
5. 私信内容 mesdes varchar 500
6. 私信时间 mesdate varchar 50
7. 私信状态 messtatus varchar 50
8. 私信其他信息：键值对 mesifno varchar 255

十三. **任务对象 Task** —— 数据库表

1. 任务主键 id taskid int 10
2. 任务发起人 taskfrom_mid int 10 一个
(源任务接收人被改为多对多)
3. 任务描述 taskdes varchar 255
4. 任务发起时间 taskstart varchar 255
5. 任务截止期限 taskdead varchar 255
6. 任务结束时间 taskend varchar 255

7. 任务状态 taskstatus varchar 50
8. 任务其他信息 taskinfo varchar 255

十四. **博客对象 Blog** —— 数据库表

1. 博客主键 id blogid int 10
2. 博客作者 blogmid varchar 255
3. 博客标题 blogtitle varchar 255
4. 博客内容 blogcont varchar 255
5. 博客发布时间 blogdate varchar 255
6. 博客状态 blogstatus varchar 255
7. 博客其他信息 bloginfo varchar 255

十五. **代码版本对象 Version** —— 数据库表

1. 版本主键 vid int 10
2. 版本父版本 vbefore int 10
3. 版本配置文件 xml 路径 vpath varchar 255
4. 版本更新日期 vdate varchar 255
5. 版本更新人 vmid int 10
6. 版本所属团队 vtid int 10
7. 版本其他信息 vinfo varchar 255

十六. **项目笔记对象 ProjectText** —— 数据库表

1. 项目笔记主键 ptid int 10
2. 项目笔记撰写人 ptmid int 10
3. 笔记对应项目 pttid int 10

4. 项目笔记标题 pttitle varchar 255
5. 项目笔记内容 ptcont varchar 255
6. 项目笔记状态 ptstatus varchar 255
7. 项目笔记其他内容 ptinfo varchar 255

十七. **日程对象 Cale** —— 数据库表

1. 日程主键 id caid int 10
2. 日程日期 cadate varchar 255
3. 日程所属项目 catid int 10
4. 日程状态 castatus varchar 50
5. 日程其他信息 cainfo varchar 255

十八. **操作过滤器对象 (Operate)** —— 数据库表

1. 操作过滤器 id
2. 操作过滤器业务
3. 操作过滤器权限集合 (多对多)

十九. **成员 (Member) —— 职位 (Post)** 多对多关系模型

Member_Post

1. 团队成员外键 mid
2. 职位外键 pid

二十. **职位——权限多对多关系模型 Post—Role**

1. 职位外键 pid
2. 权限外键 rid

二十一. **通知——团队成员 Notification—Member** 多对多关

系模型

1. 通知 id
2. 用户 id

二十二. **项目组——团队成员 Team—Member** 多对多关系模型

1. 项目组 id
2. 团队成员 id

二十三. **任务——团队成员 Task—Member** 多对多关系模型

1. 任务 id
2. 接收任务的团队成员 id

二十四. **私信——用户 Message—Member** 多对多关系模型

1. 私信 id
2. 接收私信的团队成员 id

一、模块/功能 分析阶段 有了数据之后就要开始定义模块了

a) 话说模块应该怎么定义啊？

b) 用户 (Member) 模块 (*这里必须要用到中介者模式*)

- i. 用户的增删改查业务 注册用户 登录用户 修改用户信息 封停用户
- ii. 用户的通过邮箱找回密码业务
- iii. 用户的从未验证状态转化为正常状态 查看
- iv. 用户的个人信息编辑 增删改查
- v. 用户的职位管理 查看 申请
- vi. 用户的个人私信的维护 发送 查看 删除 草稿箱
- vii. 用户的文件树的维护 增删改查

- viii. 用户写博客的维护 增删改查
- ix. 用户个人任务的维护 查看 交付
- x. 用户采购信息的维护 申请采购 审批采购 查看采购 撤销采购
- xi. 用户所属团队的维护 申请加入 申请退出 查看详情 申请建立
- xii. 用户占有的项目笔记的维护
- xiii. 用户收到的推送通知的维护 查看 删除
- xiv. 用户对版本的维护

c) 职位维护模块

- i. 添加职位
- ii. 删除职位
- iii. 修改职位基本信息
- iv. 修改职位权限信息

d) 权限维护模块

- i. 增加权限
- ii. 修改权限
- iii. 删除权限
- iv. 确定权限的职责

e) 权限过滤器模块

- i. 增加每个页面的【操作】对象，同时使用 AOP 切面方式进行校验
- ii. 符合的显示内容，不符合的直接死（弹出 503 GateWay）
- iii. 设置过滤器权限（于管理员账号）

f) 团队维护模块

- i. 由具有权限的用户进行团队的创建
- ii. 由创建者或者管理员进行团队信息的更改
- iii. 由具有权限的用户进行版本库的迭代与创建新的基类版本
- iv. 对团队成员的维护（邀请，剔除、管理权限）
- v. 项目组的日程的管理
- vi. 项目组的项目文档的管理
- vii. 项目组文件树的管理

g) 通知模块

- i. 也是介乎于用户之间的 由某个管理员权限的人进行整个特殊通知的发送
- ii. 用户对于收到的通知的状态变更，即未读、已读
- iii. 查看所有的通知
- iv. 删除一些通知

h) 私信模块

- i. 用户与用户之间的私信传递
- ii. 对于发送人来说 发信、草稿箱、撤回业务（草稿箱，已发送，已接收）
- iii. 对收到的信息 信的状态更改（已读，以删除）

i) 用户审批模块

- i. 某个用户提出申请，或者代码提交等，自动产生审批操作
- ii. 具有审批权限的用户进行审批的处理（同意和拒绝）

j) 任务维护模块

- i. 由高权限者向低权限者给予任务
 - ii. 通过提交等信息来进行任务状态的变更(List)
 - iii. 拒绝任务等操作
- k) 文件树维护模块
 - i. 对文件树进行添加文件
 - ii. 删除某个文件
 - iii. 文件夹的添加和删除
 - iv. 不能直接修改，只能覆盖和下载像是 HDFS 那样的感觉
- l) 新闻编写模块
 - i. 管理员进行增删改查
- m) 博客维护模块
 - i. 用户自己进行博客的编写
- n) 采购维护模块
 - i. 同审批模块
- o) 版本维护模块
 - i. 这个待定

二、页面布局设计（分为几个界面）

- a) 团队成员（Member）注册+登录页面（已设计）
- b) 通用模板 template.jsp
 - i. 设计 head（css 引入）--完成
 - ii. 设计 header（顶部栏）
 - 1. 显示推送数量+详情

2. 显示通知数量+详情

3. 显示私信数量+详情

- iii. 设计 footer (尾标)

1. 版权声明而已

- iv. 设计 sidebar (菜单栏) 重点设计

1. 欢迎——主界面，涵盖大部分信息

2. Dsa

3. 用户办公模块

- a) 每日签到

- b) 职位维护

- c) 私信维护

- d) 任务维护

- e) 审批维护

- f) 推送通知维护

4. 用户维护模块 (我还没想好叫啥)

- a) 维护个人信息

- b) 修改个人密码

- c) 网站个人偏好设置

- v. 设计 js (js 引入)

- vi. 设计 style (主栏目的设置)

三、具体实现阶段

四、单元测试阶段

五、模块测试阶段

六、压力测试阶段

七、试上线