## 文案星球系统分析与设计成果报告

班级 1915432 学号 191543205 姓名 李淑仪 得分 \_\_\_\_\_\_\_\_\_\_

其它组员：学号 191543203 姓名 黄文宝

**主要工作：**

李淑仪： 设计和完成前端页面，设计数据库，实现前后端交互，实现部分控制层接口，页面测试，文档编写。

黄文宝： 提供产品需求，创建数据库，设计和实现持久层、service层，控制层开发，实现大部分接口，接口测试。

整个项目gihub网址：

#### 1选题

1．选题名称

**文案星球 -- 专注于分享文案的平台**

2．选题描述

现在发现很多网站都有类似朋友圈这种发表功能，但是这些网站中，有一个共同的点就是功能太多太杂，不能专注于做一件事情。所以本网站的主要痛点就是，主题明确，功能少而精。围绕“文案分享”中的文案设计产品需求，在本网站中只发表于文案有关的内容，用户能浏览能操作的东西只有文案。根据这个需求，我们项目的主要功能就是发表文案，欣赏文案，收藏文案，举报文案。

#### 2用户需求

2.1 功能需求

（1）用户管理

* 用户信息注册，注册用户为游客。系统默认提供一名管理员用户。管理员可对已注册的用户行使管理职权，如审核用户发表内容、禁言用户、注销用户账号等。
* 普通用户注册后，可以查看“文案广场”的文案、对感兴趣的文案进行，收藏和分享、可以发表喜欢的文案到“文案广场”、可以举报其他文案、可以注销自己的账号等
* 普通用户在登录后均可修改个人信息。
* 未注册的游客可进入平台主页，浏览平台的公开内容，也可进入注册或登录页面。

（2）文案发表

* 用户进入发表文案页面，编辑文案，进行发表
* 经过管理员审核无敏感内容后，文案将成功发表存入系统
* 用户可以在个人资料页面，看到自己的历史文案，也可以删除已发表过的文案

（3）文案点赞、收藏

* 用户在文案广场浏览文案的同时，可以对自己喜欢的文案进行点赞或者收藏操作
* 收藏会将本文案放进用户的收藏夹里
* 用户可以在收藏夹里查看收藏过的文案以及对其中的文案进行取消收藏操作
* 如果是游客，触发了收藏操作后，会提示让注册/登录

（4）举报用户

* 用户可以在看到较敏感的文案时，对该文案进行举报
* 管理员看到举报信息，举报的内容有问题的话，就警告发表该内容的用户；反之，警告举报的用户；警告以公告的形式发到首页里

（5）用户管理

* 一个用户被举报的次数过了一定的量时，对该用户进行禁言/销户操作

2.2 用例模型

（1）确定参与者

主要参与者包括：游客、普通用户、管理员。

（2）确定用例

根据系统功能的描述，可以分离出的主要用例包括：注册，登录，修改个人信息，浏览文案，发表文案，分享文案，查看发表过的文案，删除发表过的文案，查看收藏的文案，收藏文案，举报用户，处理举报信息，发表公告，禁言用户，注销用户等。

（3）确定用例和参与者的关系

游客可以浏览平台公开信息，注册。

普通用户除了作为游客身份使用系统外，还可以登录，修改个人信息，浏览文案，发表文案，收藏文案，分享文案，查看发表过的文案，删除发表过的文案，查看收藏的文案，取消收藏的文案，举报用户。

管理员除了作为用户身份使用系统外，还可以登录，用户管理，处理举报信息，发表系统公告，禁言用户，注销用户。

2.2.1 用例图

要求与功能需求一致



2.2.2用例模型说明

“注册”用例文档

|  |  |
| --- | --- |
| **用例名** | 注册 |
| **简要描述** | 游客利用该用例注册系统，身份认证信息存入系统，游客身份转变为客户（管理员非注册得到，而是系统内定账号密码输入） |
| **参与者** | 游客 |
| **涉众** | 游客：个人信息保存到系统，从游客身份转换到用户，并获得相应的权限 |
| **相关用例** | 登录 |
| **前置条件** | 个人信息未存入系统中，即未注册过 |
| **后置条件** | 无 |
| **基本事件流** | 1. 用例起始于用户需要注册使用该系统 2. 系统显示欢迎界面，并要求用户选择注册或登录 3. 用户输入用户名和密码 4. 系统存入用户名和密码，用户直接登录系统 |
| **补充约束-数据需求** | 1. 邮箱要满足Email的格式要求 2. 用户名不能跟系统存在的重复 |
| **补充约束-业务规则** | 1. 注册账号已存在，提示信息，回到欢迎页面 |
| **待解决问题** | 无 |
| **相关图** | 可提供注册过程的活动图 |

“登录”用例文档

|  |  |
| --- | --- |
| **用例名** | 登录 |
| **简要描述** | 用户和管理员利用该用例登录系统，通过身份认证获得相应的操作权限 |
| **参与者** | 用户和管理员 |
| **涉众** | 用户和管理员：通过身份验证，并获得相应的权限 |
| **相关用例** | 注册 |
| **前置条件** | 用户在系统注册过 |
| **后置条件** | 如果登录成功，则显示相应权限的操作界面 |
| **基本事件流** | 1. 用例起始于用户需要登录到该系统 2. 系统显示欢迎界面（注册或登录），用户选择登录，页面要求用户输入用户id和密码 3. 用户输入用户id和密码 4. 系统验证用户名和密码，允许用户登录系统 5. 系统根据用户类型启动不同主操作界面，若验证不通过，则返回登录页面 |
| **补充约束-数据需求** | 无 |
| **补充约束-业务规则** | 1. 若连续3次登录失败，则锁定账户30分钟，并存储审计信息，供管理员查看 2. 登录账号不存在，提示信息，回到欢迎页面 |
| **待解决问题** | 无 |
| **相关图** | 可提供登录过程的活动图 |

“销户”用例文档

|  |  |
| --- | --- |
| **用例名** | 注销账户 |
| **简要描述** | 用户进入个人中心页面，选择销户，系统清除用户信息 |
| **参与者** | 用户 |
| **涉众** | 用户：注销账户后，无法使用客户操作 |
| **相关用例** | 登录 |
| **前置条件** | 系统有该信息，且该用户登录成功 |
| **后置条件** | 注销账号后，将无法用该账号登录网站，下次需要使用操作需要重新注册 |
| **基本事件流** | 1. 用例起始于用户需要注销账户 2. 用户进入个人中心页面，选择销户修改成功后，系统关于改用户所有信息都清除 |
| **补充约束-数据需求** | 无 |
| **补充约束-业务规则** | 无 |
| **待解决问题** | 无 |
| **相关图** | 可提供销户过程的活动图 |

“修改个人信息”用例文档

|  |  |
| --- | --- |
| **用例名** | 修改个人信息 |
| **简要描述** | 用户进入个人中心页面，编辑修改个人信息，修改后的个人信息存入系统 |
| **参与者** | 用户，管理员 |
| **涉众** | 用户，管理员：拥有修改头像，姓名，性别等的权限 |
| **相关用例** | 登录 |
| **前置条件** | 登录成功 |
| **后置条件** | 无 |
| **基本事件流** | （1）用例起始于用户需要修改个人信息  （2）系统进入个人中心页面，并根据用户操作进行信息修改  （3）修改成功后，新的信息存入系统进入数据库 |
| **补充约束-数据需求** | 无 |
| **补充约束-业务规则** | 无 |
| **待解决问题** | 无 |
| **相关图** | 可提供修改个人信息过程的活动图 |

“发表文案”用例文档

|  |  |
| --- | --- |
| **用例名** | 发表文案 |
| **简要描述** | 用户进入发布页面编辑发表，记录进系统，可在个人中心“创作”模块中展示和进行后续操作（删除） |
| **参与者** | 用户 |
| **涉众** | 用户，管理员：发布选择公开后的文案可以被所有人看到 |
| **相关用例** | 欣赏文案 |
| **前置条件** | 登录成功 |
| **后置条件** | 发表文案成功后，用户可以在“文案广场”和个人中心的历史文案查看，也可以对已发表过的文案进行删除 |
| **基本事件流** | 1. 用例起始于发布页面 2. 用户选择文案类型，编辑文案并上传 3. 用户在文案广场可以看见自己发表过的文案 4. 个人中心的历史文案查看，也可以对已发表过的文案进行删除 |
| **补充约束-数据需求** | 1. 含有敏感词被检测到将会发布失败，需要修改 |
| **补充约束-业务规则** | 无 |
| **待解决问题** | 无 |
| **相关图** | 可提供发表文案的活动图 |

“欣赏文案”用例文档

|  |  |
| --- | --- |
| **用例名** | 欣赏文案 |
| **简要描述** | 用户和游客在文案广场欣赏所有已发表过的文案，可以对文案进行收藏操作，在个人中心收藏文案里会有记录，对于一些过于敏感的文案可以进行举报 |
| **参与者** | 用户 |
| **涉众** | 用户，游客，管理员：发布选择公开后的文案可以被所有人看到 |
| **相关用例** | 发表文案 |
| **前置条件** | 登录成功 |
| **后置条件** | 无 |
| **基本事件流** | 1. 用例起始于世界页面 2. 用户可以在文案广场欣赏文案并进行相关操作 3. 游客只可以看到文案，但不能进行操作记录 4. 用户在个人中心相关模块看到自己的操作记录 |
| **补充约束-数据需求** | 无 |
| **补充约束-业务规则** | 无 |
| **待解决问题** | 无 |
| **相关图** | 可提供欣赏行为的活动图 |

“收藏文案”用例文档

|  |  |
| --- | --- |
| **用例名** | 收藏文案 |
| **简要描述** | 用户进入文案广场页面，在欣赏文案的同时，可对感兴趣的文案进行收藏，然后在个人中心查看已收藏过的文案，也可以对不敢兴趣的已收藏文案，进行取消收藏 |
| **参与者** | 用户 |
| **涉众** | 用户，游客，管理员：发布选择公开后的文案可以被所有人看到 |
| **相关用例** | 欣赏文案 |
| **前置条件** | 登录成功，文案未被本用户收藏 |
| **后置条件** | 文案收藏成功后，用户可以在个人中心的文案收藏里查看已经收藏过的文案，也可以对文案进行取消收藏 |
| **基本事件流** | （1）用例起始于文案广场页面   1. 用户可以对任一文案进行收藏操作   （3）用户在个人中心看到自己的收藏过的文案，可以对其进行取消收藏 |
| **补充约束-数据需求** | 无 |
| **补充约束-业务规则** | 无 |
| **待解决问题** | 无 |
| **相关图** | 可提供收藏文案的活动图 |

、

“举报”用例文档

|  |  |
| --- | --- |
| **用例名** | 举报用户 |
| **简要描述** | 用户对敏感的文案进行举报，管理员对被举报的信息进行审核并警告举报者或者被举报者 |
| **参与者** | 管理员 |
| **涉众** | 用户，管理员 |
| **相关用例** | 欣赏文案 |
| **前置条件** | 管理员登录成功，用户登录成功 |
| **后置条件** | 无 |
| **基本事件流** | 1. 用例发起于举报页面 2. 用户在看到敏感内容，就可以对该内容进行举报 3. 管理员在看到举报信息后，审核被举报内容，将审核无误的被举报内容删除并对用户进行警告处理；反之，对随意举报的用户进行警告处理 |
| **补充约束-数据需求** | 无 |
| **补充约束-业务规则** | 无 |
| **待解决问题** | 无 |
| **相关图** | 可提供发表文案的活动图 |

“发布公告”用例文档

|  |  |
| --- | --- |
| **用例名** | 发布公告 |
| **简要描述** | 管理员在发布公告页面，发表警告/系统更新信息 |
| **参与者** | 管理员 |
| **涉众** | 管理员 |
| **相关用例** | 无 |
| **前置条件** | 管理员身份登录 |
| **后置条件** | 无 |
| **基本事件流** | 1. 用例起始于发布公告页面 2. 管理员选择公告类型，编辑公告内容并发布 3. 游客，用户，管理员对能在首页看到公告内容 |
| **补充约束-数据需求** | 无 |
| **补充约束-业务规则** | 无 |
| **待解决问题** | 无 |
| **相关图** | 可提供发布公告的活动图 |

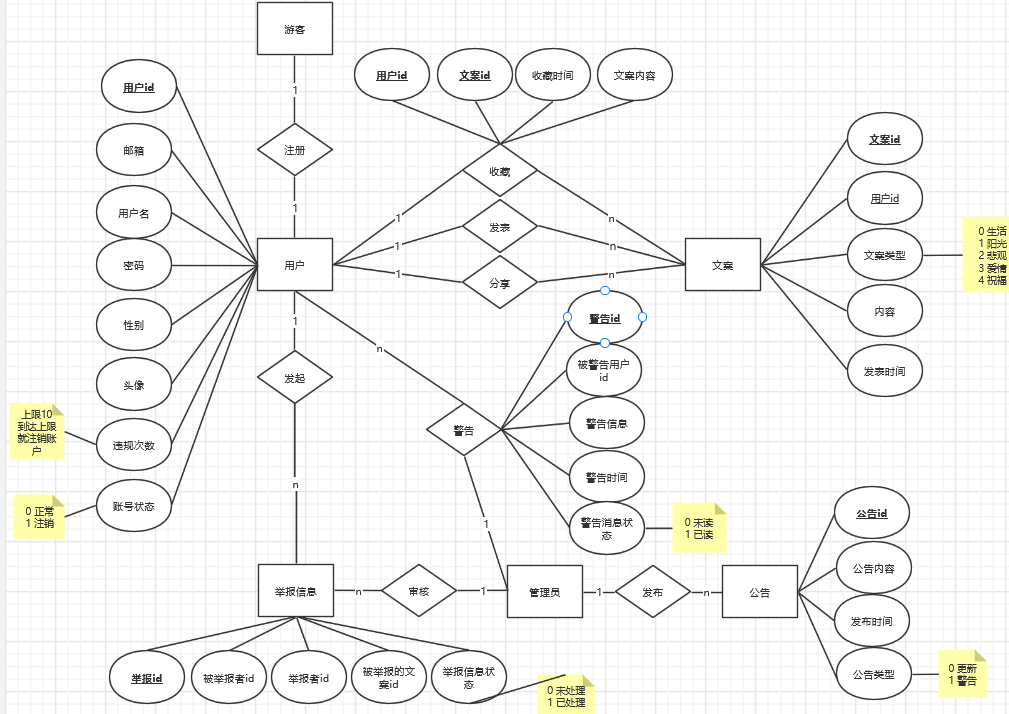
“社区维护”用例文档

|  |  |
| --- | --- |
| **用例名** | 社区维护 |
| **简要描述** | 管理员进行用户管理，审核文案，图片，对不恰当的内容进行删除警告操作，对违规用户进行强制销户 |
| **参与者** | 管理员 |
| **涉众** | 客户，管理员：管理员对于用户的行为规范，在自由的前提保持社区的良好 |
| **相关用例** | 无 |
| **前置条件** | 管理员身份登录 |
| **后置条件** | 无 |
| **基本事件流** | 1. 用例起始于发布页面 2. 用户选择频道编辑文案并上传 3. 用户在相关频道看见自己编辑的文案 4. 用户在个人中心看到自己的文案，可以对自己的文案进行重新编辑，转权限和删除等相关操作 |
| **补充约束-数据需求** | 无 |
| **补充约束-业务规则** | 无 |
| **待解决问题** | 无 |
| **相关图** | 可提供发表文案的活动图 |

#### 3数据库设计

3.1 概念结构设计

E-R图及其说明

 3.2 逻辑结构设计

关系模式

用户：user(uid,uemail,uname,upwd,usex,uface,uillegalTime,ustatus)

文案：copywirting(cwid,uid,cwtext,cwtype,cwtime)

收藏：collect(uid,cwid,ctime,cwtext)

公告：notion(nid,ncontext,ntime,ntype)

举报：inform(inid,ineduid,inuid,cwid,instatus)

警告：warn(wid,uid,ucontext,wtime,wstatus)

3.3 数据库实现SQL代码

基于MySQL数据库给出全部的SQL脚本

|  |
| --- |
| -- 创建数据库 cwenjoy  create database cwenjoy;  use cwenjoy;  -- 用户  create table user(  uid int(11) primary key auto\_increment,  uemail varchar(30) not null,  uname varchar(10) not null,  upwd varchar(16) not null,  usex char(2) not null,  uface varchar(100) not null,  uillegalTime char(2) not null,  ustatus char(2) not null  ) ;  -- 文案  create table copywriting(  cwid int primary key auto\_increment,  uid int(11) not null,  cwtext varchar(200) not null,  cwtype char(2) not null,  cwtime timestamp not null,  foreign key(uid) references user(uid)  )  -- 收藏  create table collect(  uid int(11) not null,  cwid int not null,  ctime timestamp not null,  cwtext varchar(200) not null,  foreign key(uid) references user(uid),  foreign key(cwid) references copywriting(cwid)  )  -- 公告  create table notion(  nid int primary key auto\_increment,  ncontext varchar(100) not null,  ntime timestamp not null,  ntype char(2) not null  )  -- 举报  create table inform(  inid int primary key auto\_increment,  ineduid int(11) not null,  inuid int(11) not null,  cwid int not null,  instatus char(2) not null,  foreign key(ineduid) references user(uid),  foreign key(inuid) references user(uid),  foreign key(cwid) references copywriting(cwid)  )  --警告  create table warn(  wid int primary key auto\_increment,  uid int(11) not null,  wcontext varchar(100) not null,  wtime timestamp not null,  wstatus char(2) not null,  foreign key(uid) references user(uid)  ) |

#### 4类设计

4.1 主要系统类、功能及分类

- src 源码

    - main

        - java.com.lsy.cw                        项目代码

            - controller                             控制层代码

                - CollectController.java                 收藏控制类

                - CopywritingController.java              文案控制类

                - InformController.java                   举报控制类

                - NotionController.java                   公告控制类

                - UserController.java                     用户控制类

                - WarnController.java                     警告控制类

            - dao                                     dao层代码

                - CollectDao.java                         收藏Dao

                - CopywritingDao.java                     文案Dao

                - InformDao.java                          举报Dao

                - NotionDao.java                          公告Dao

                - UserDao.java                            用户Dao

                - WarnDao.java                            警告Dao

            - pojo                                     持久层代码

                - Collect.java                            收藏

                - Copywriting.java                        文案

                - Inform.java                             举报

                - Notion.java                             公告

                - User.java                               用户

                - Warn.java                               警告

            - service                                   业务层代码

                - impl                                    业务层实现类

                    - CollectServiceImpl.java              收藏业务实现类

                    - CopywritingServiceImpl.java          文案业务实现类

                    - InformServiceImpl.java               举报业务实现类

                    - NotionServiceImpl.java               公告业务实现类

                    - UserServiceImpl.java               用户业务实现类

                    - WarnServiceImpl.java                 警告业务实现类

                - CollectService.java                    收藏业务接口

                - CopywritingService.java                文案业务接口

                - InformService.java                     举报业务接口

                - NotionService.java                     公告业务接口

                - UserService.java                       用户业务接口

                - WarnService.java                       警告业务接口

            - config                                     配置

                - CharsetConfig.java                        解决乱码问题

        - resources                                   资源文件

            - static                                     前端静态资源文件

            - vue                                        前端页面

            - application.yml                            项目配置文件

    - test 测试代码

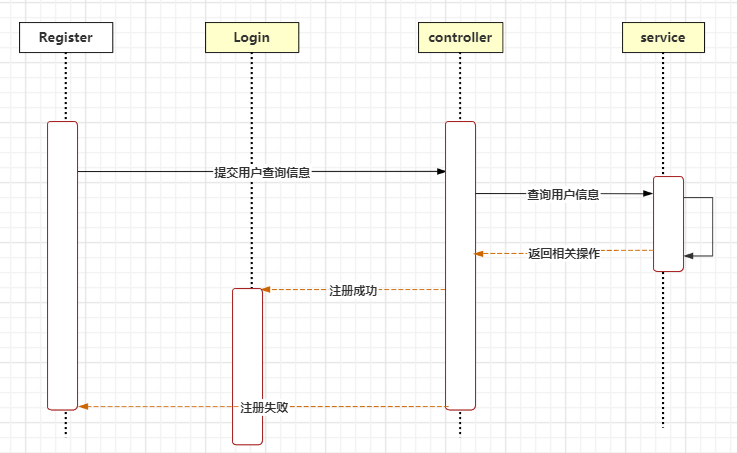
        - java.com.lsy.cw                             测试包

            - cwApplicationTests.java                     测试类

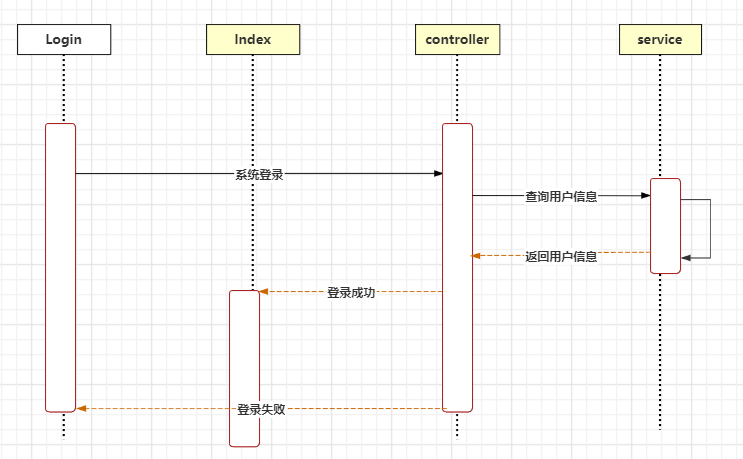
#### 5业务流程顺序图

提供用例模型中每个用例的顺序图

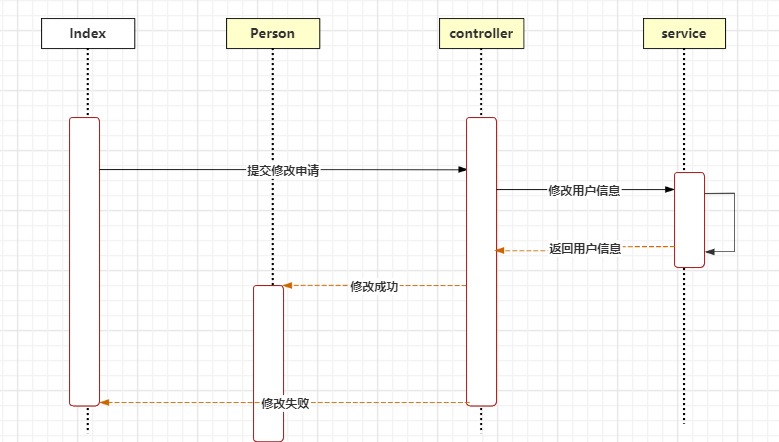
用户注册：



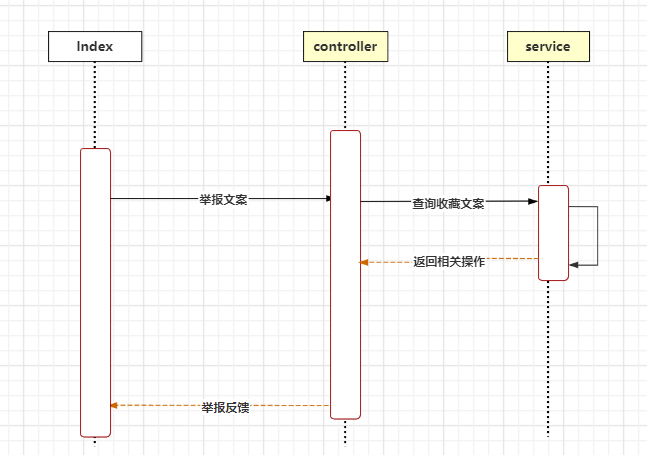
用户登录：



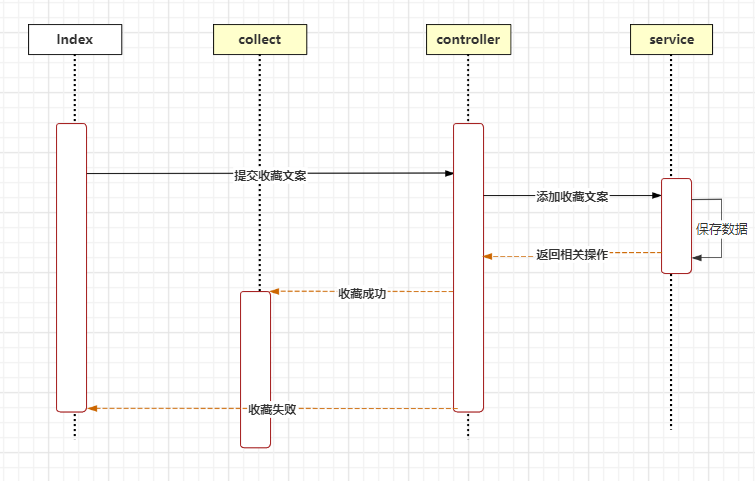
修改个人信息



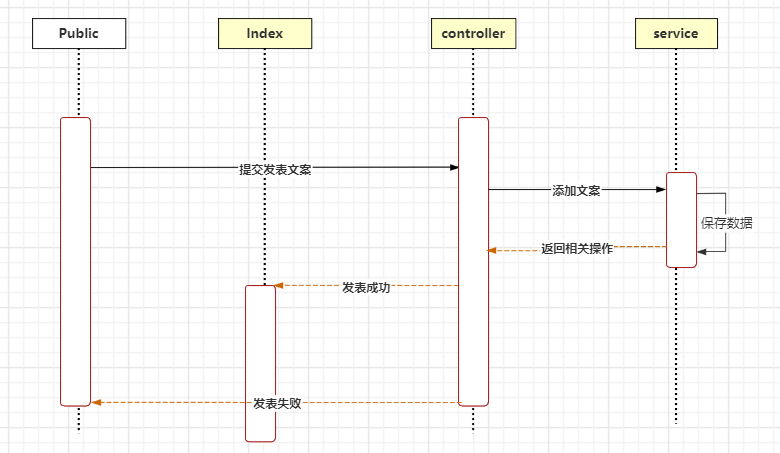
举报：



收藏文案

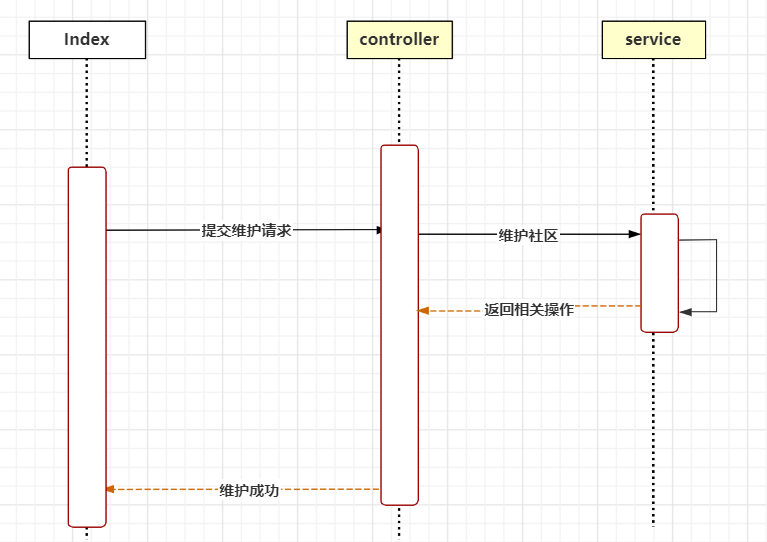


发表文案

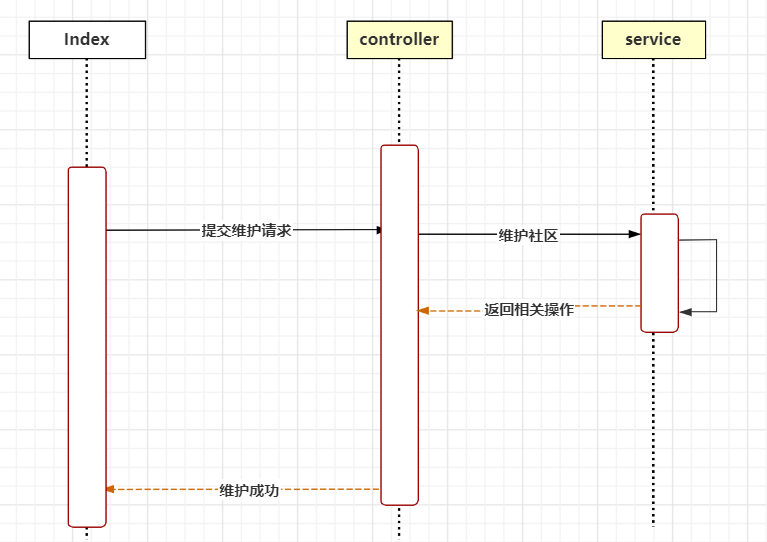


发布公告：

社区维护：



社区维护：



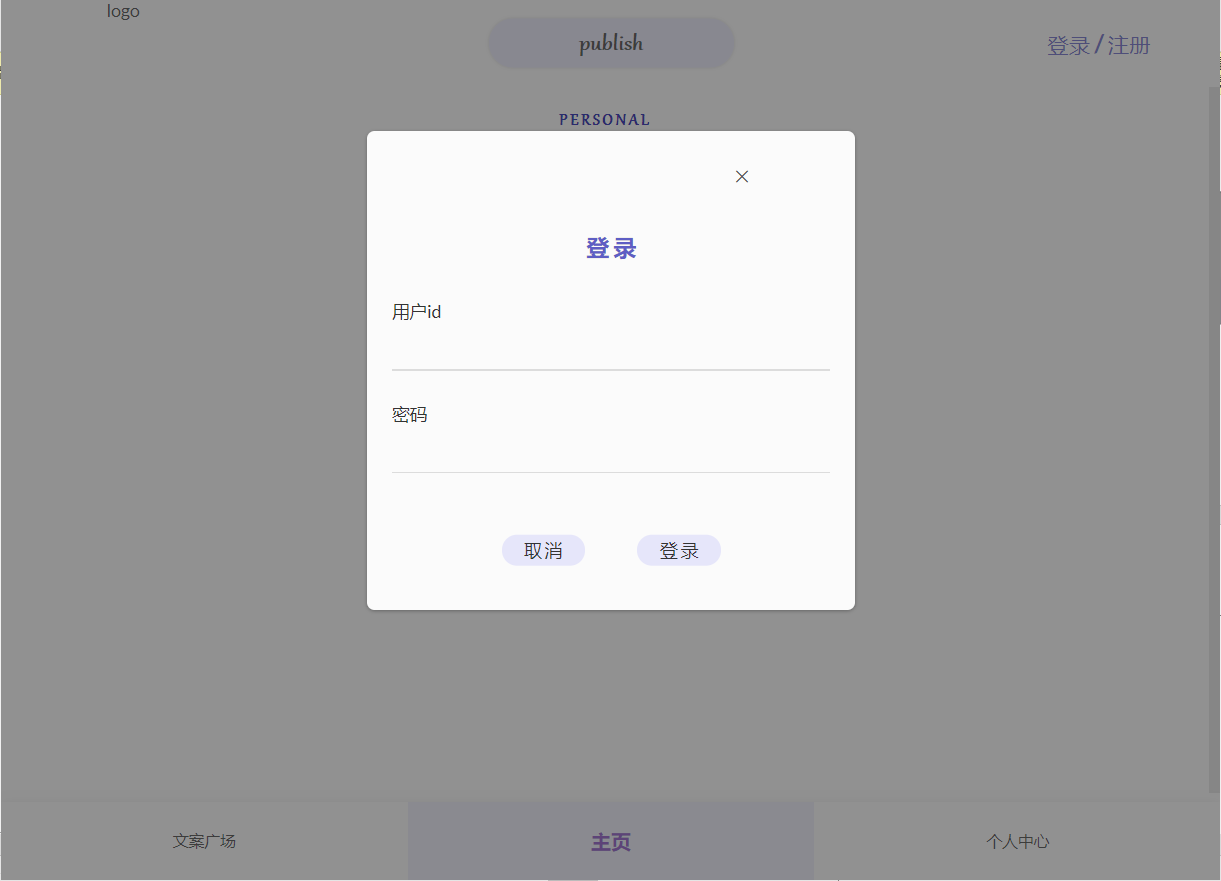
#### 6系统各模块运行界面截图

**6.1 用户**

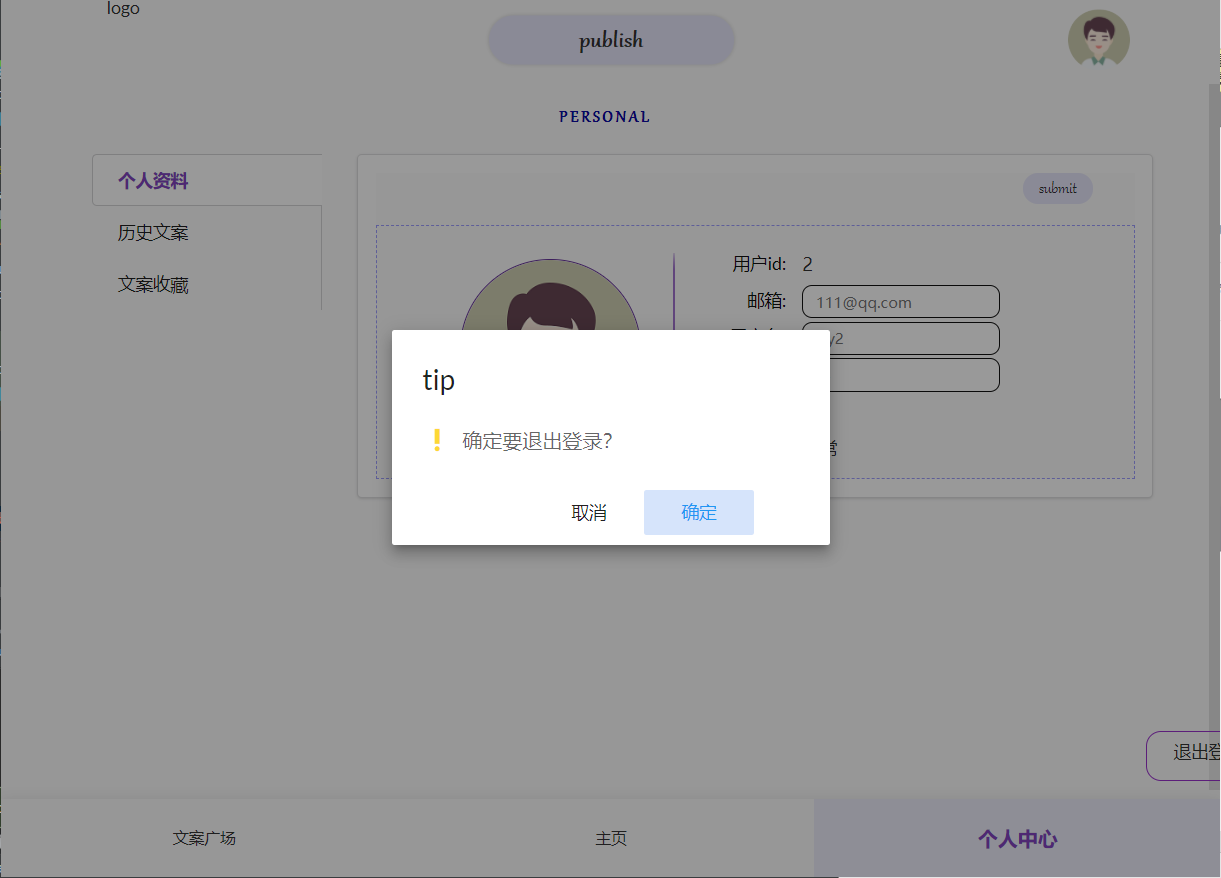
**6.1.1 注册**



**6.1.2 登录**

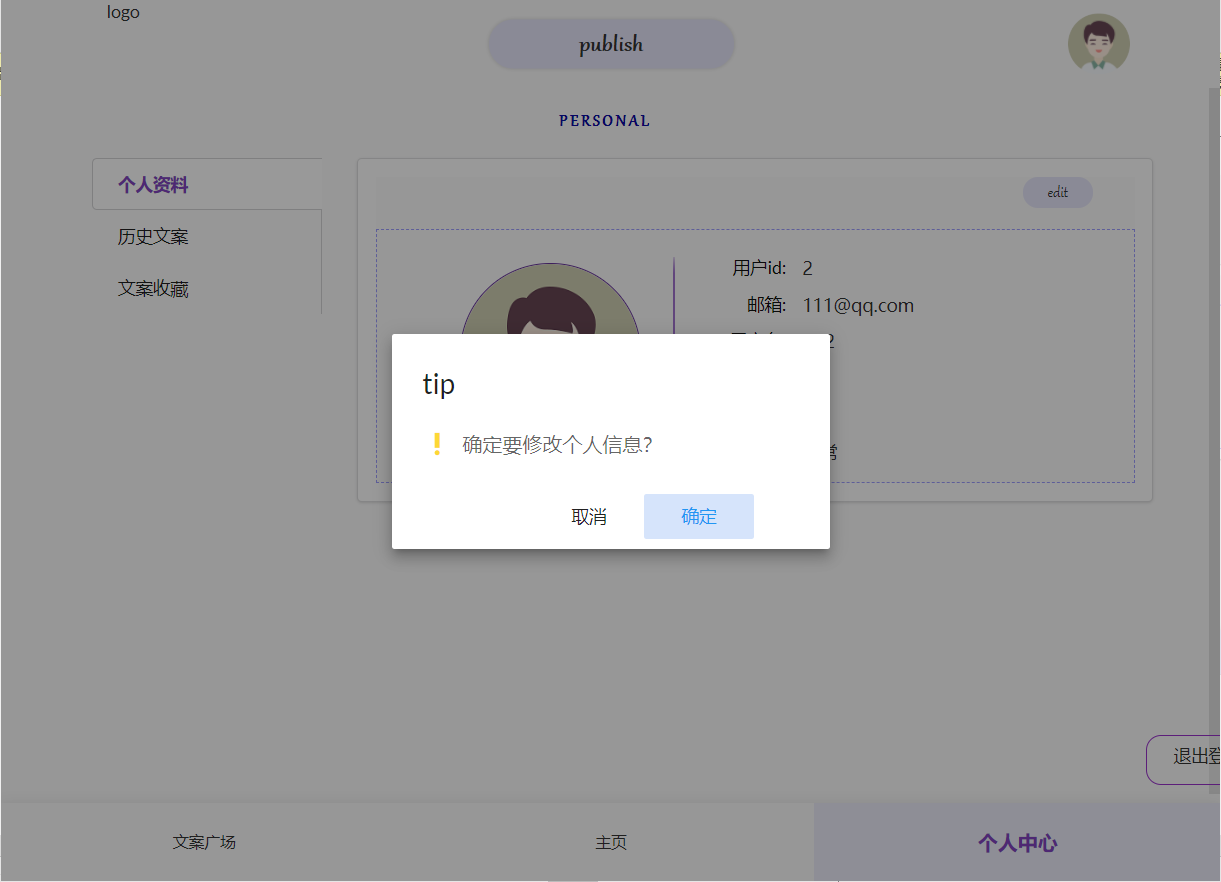


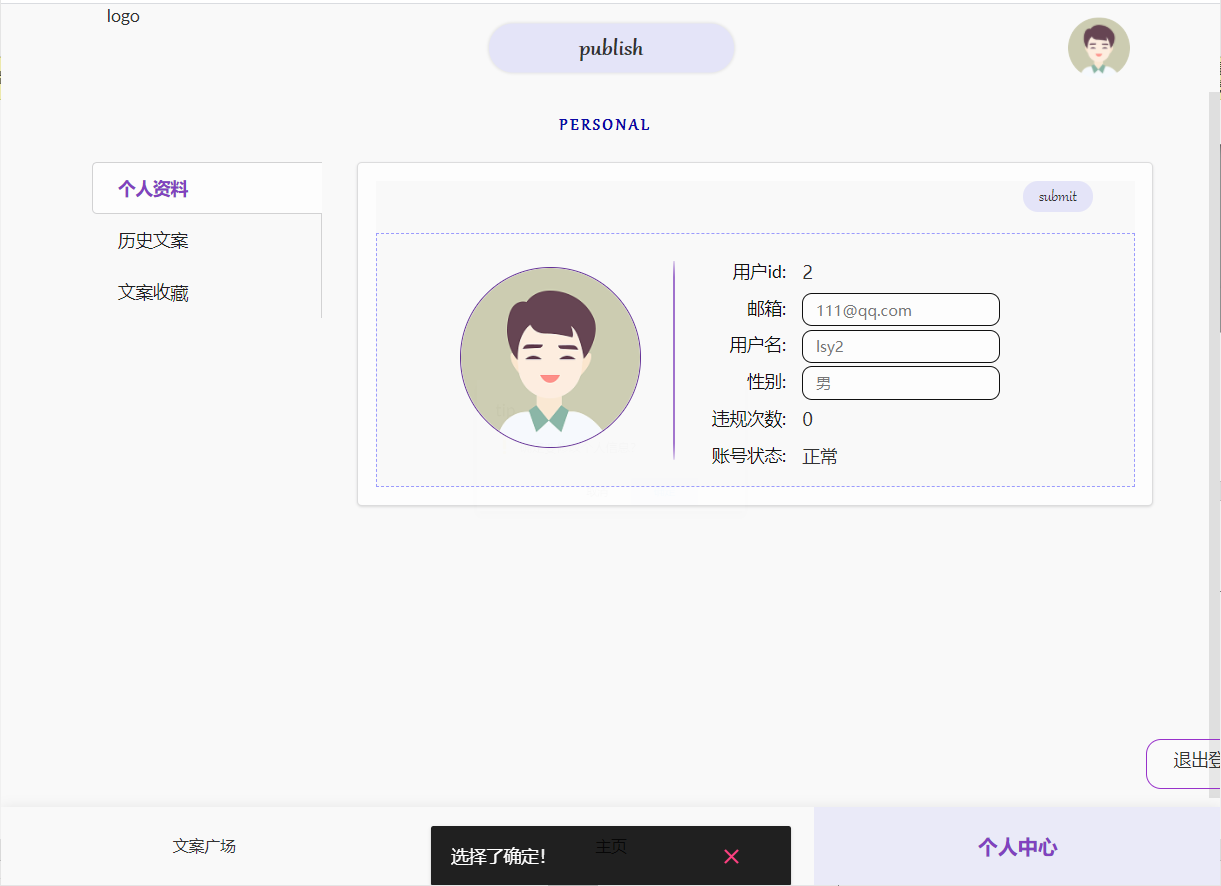
**6.1.3 退出登录**





**6.1.4 修改个人信息**





**6.2 文案**

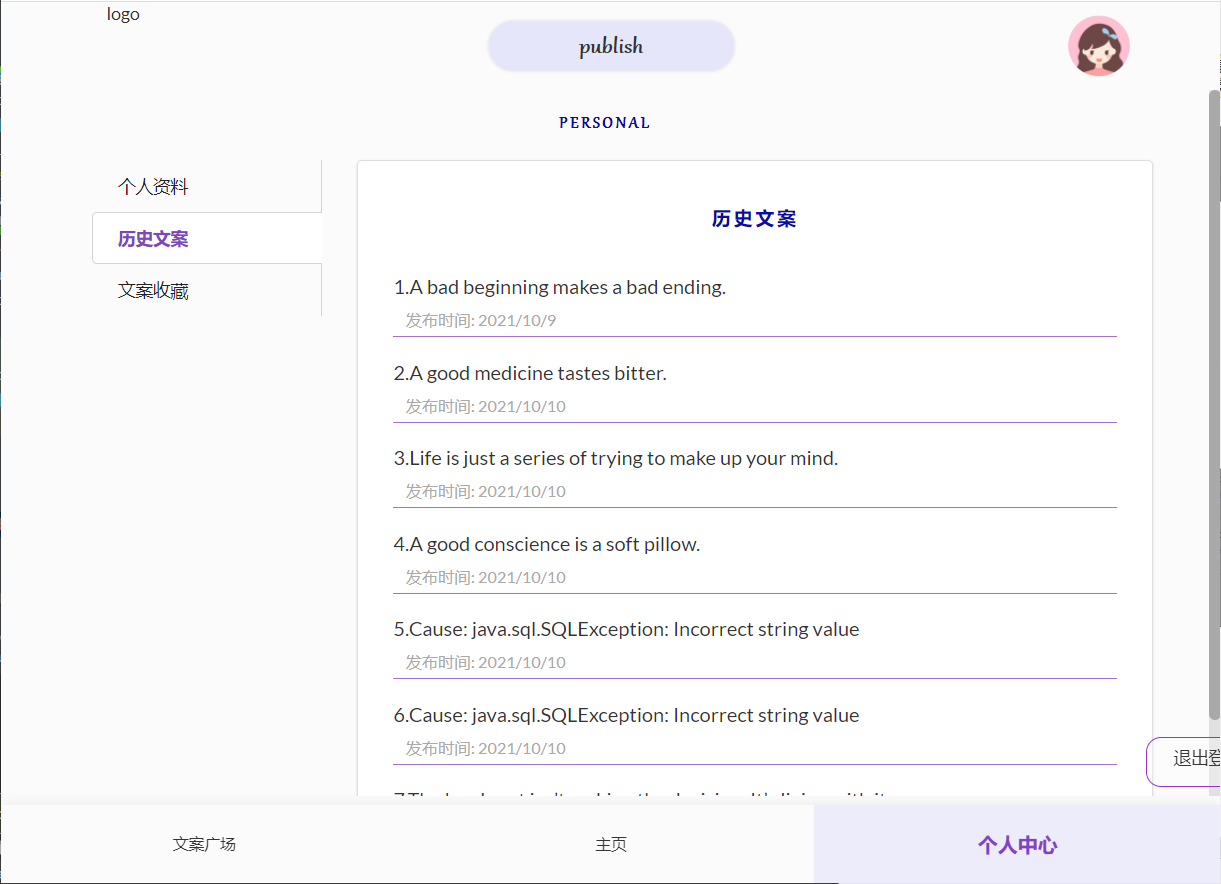
**6.2.1 文案欣赏**



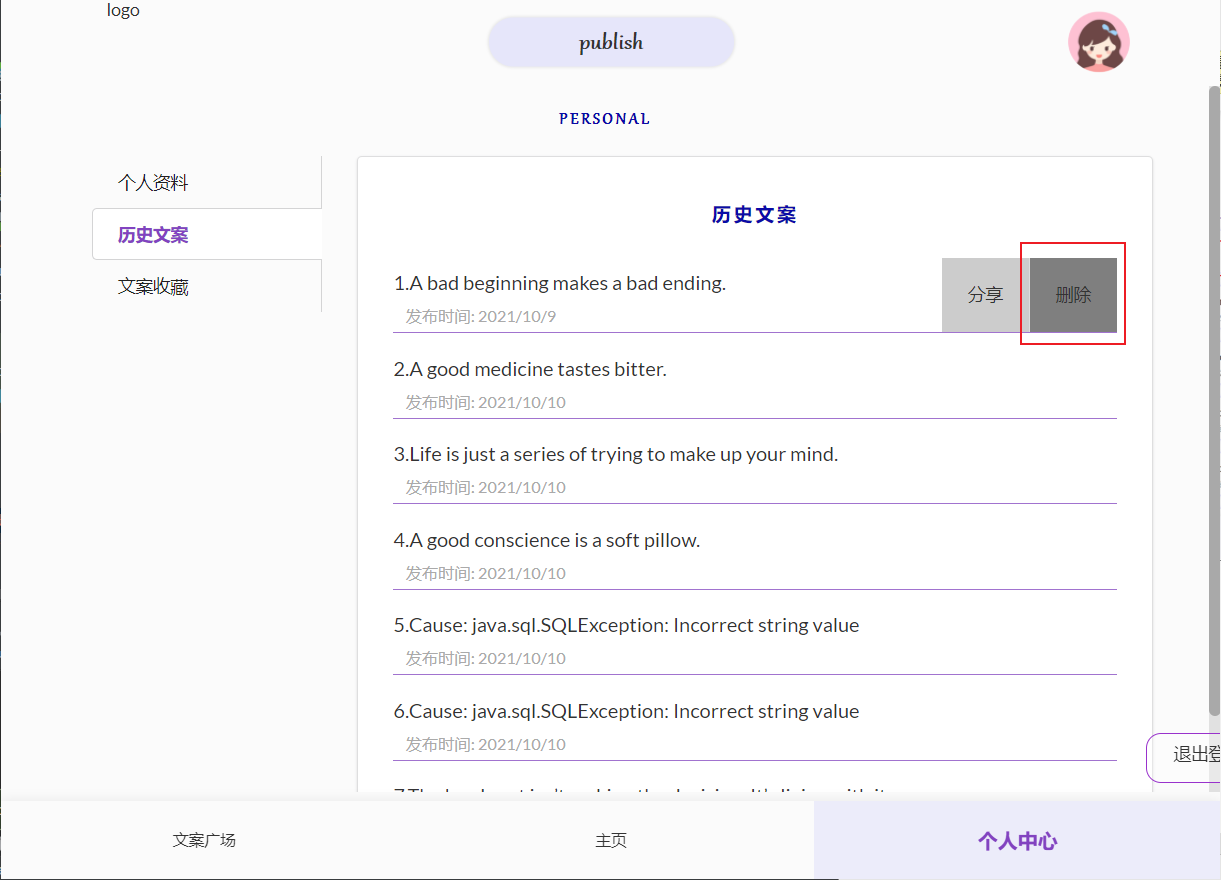
**6.2.2 发表文案**



**6.2.3 历史文案**

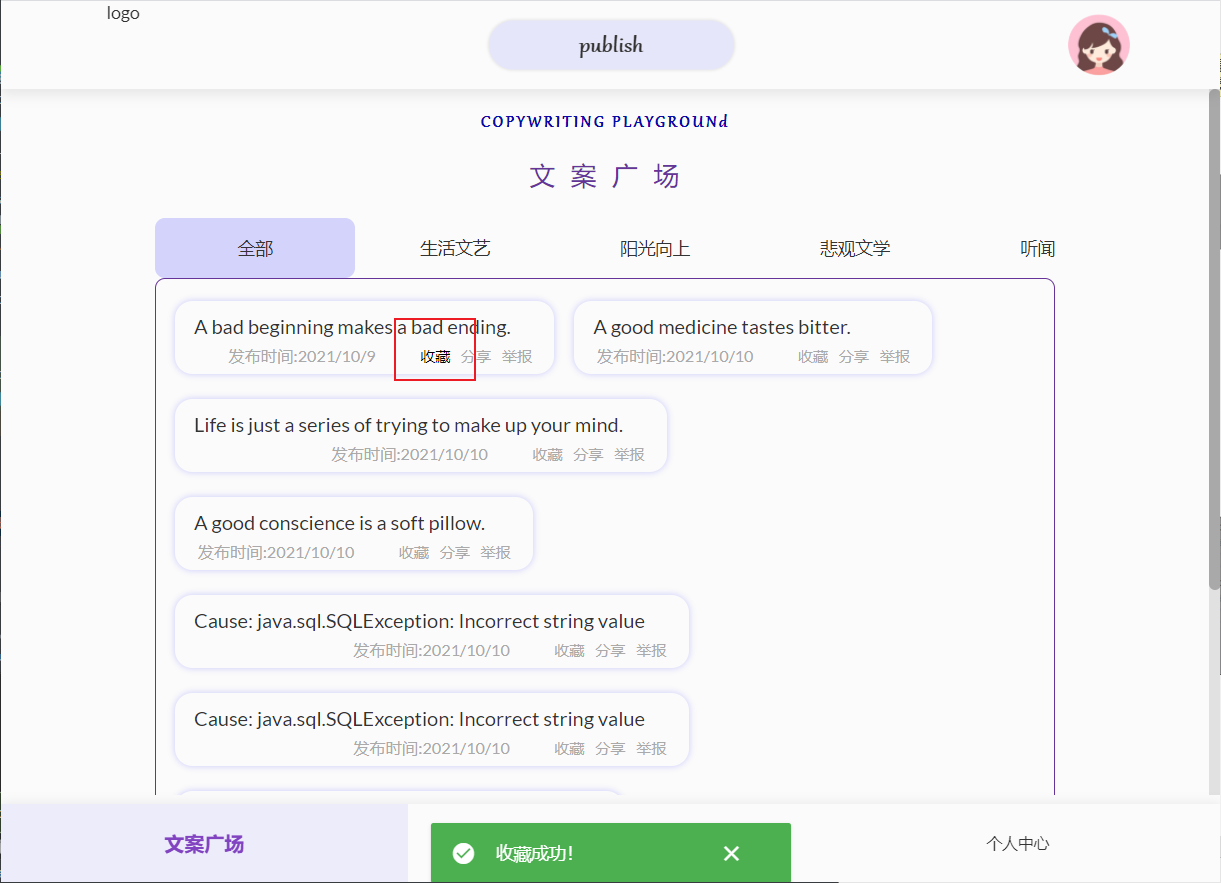


**6.2.4 删除历史文案**

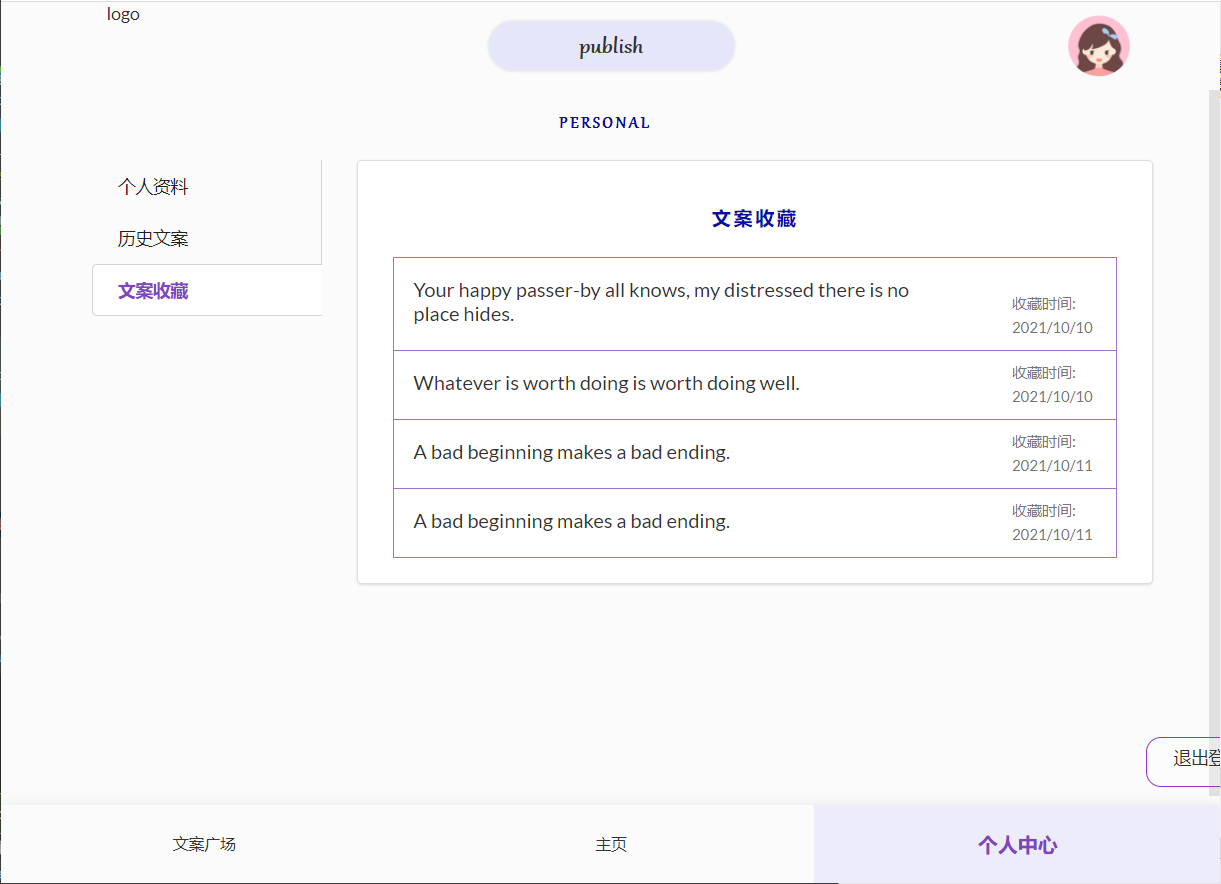


**6.3 收藏**

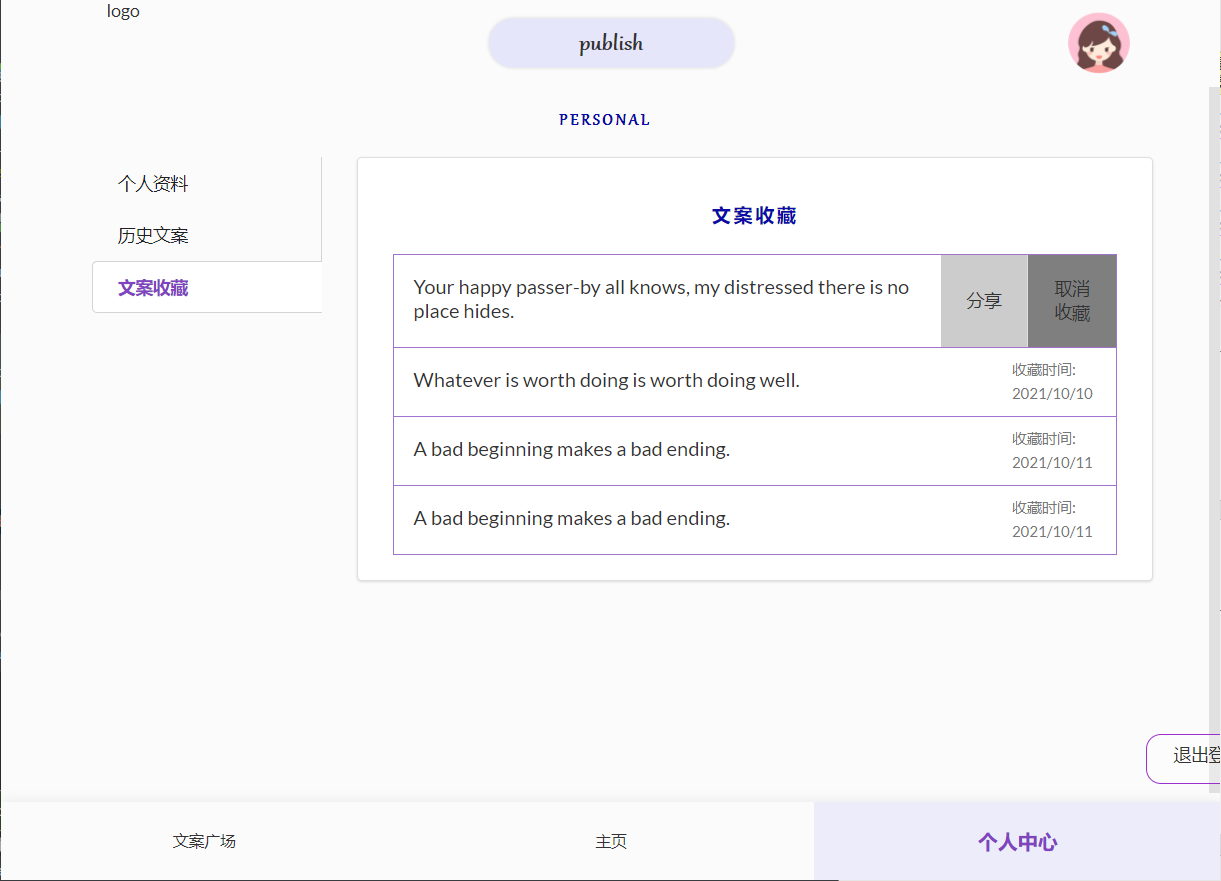
**6.3.1 收藏文案**



**6.3.2 历史收藏文案**



**6.3.3 取消历史收藏**

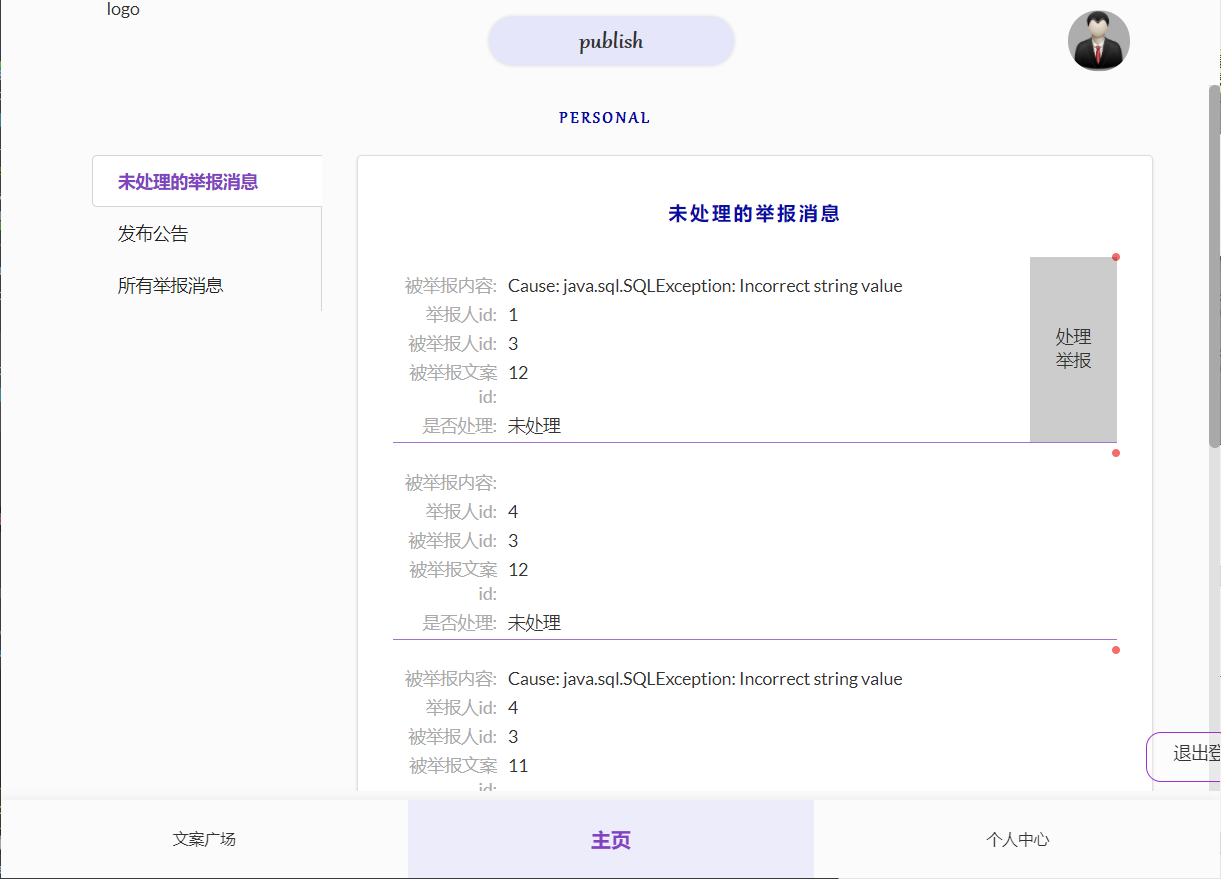


**6.4 举报**

**6.4.1 举报文案**



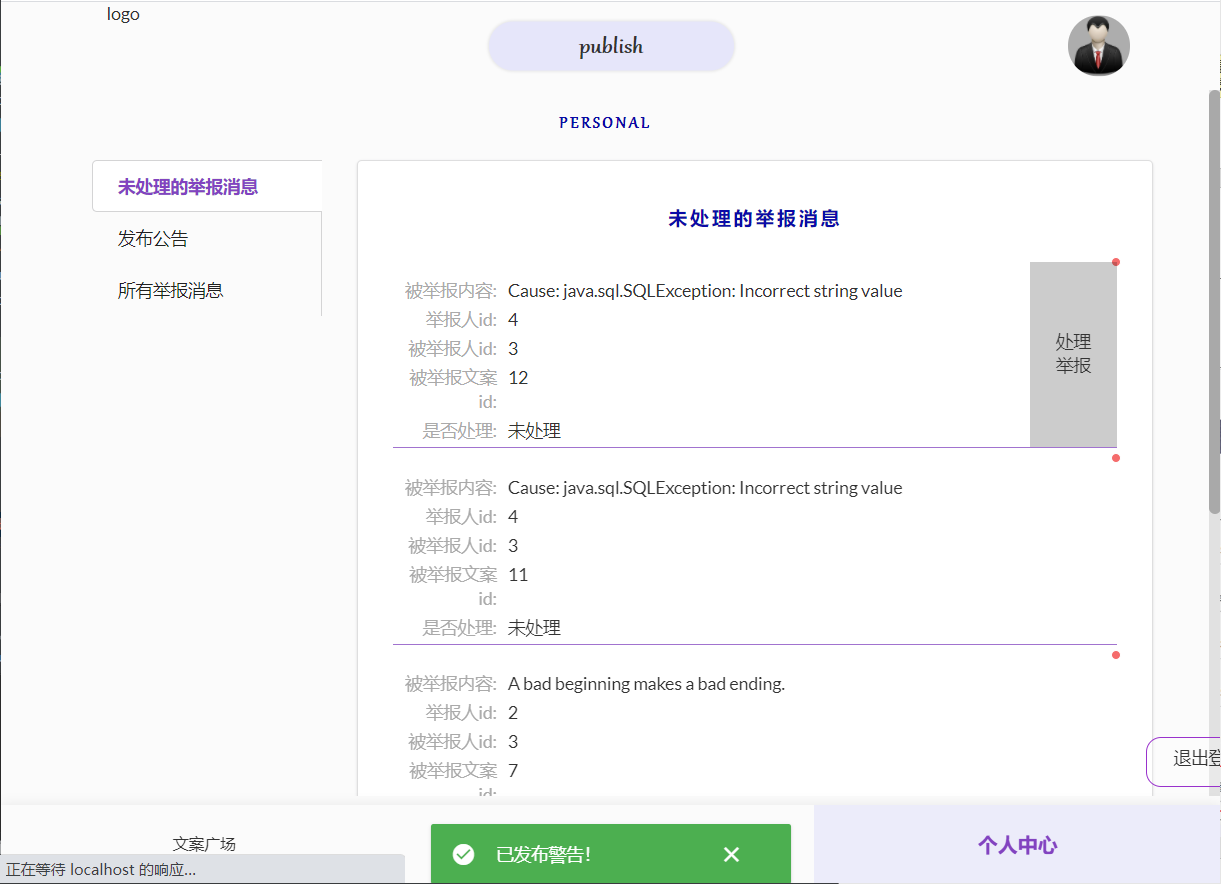
**6.4.2 管理员处理举报**





**6.4.3 发布警告**





**6.5 公告**

**6.5.1 查看公告**



**6.5.2 管理员发布公告**



