

软件设计说明书

项目名称：SchooledIn App

编写人：路萍

编写人学号：141220074

编写日期：2017.7.12

文档编号：111123232333

目录

- a. 引言 3
 - a.1 目的 3
 - a.2 修订历史 3
 - a.3 预期的读者和阅读建议 3
 - a.4 产品的范围 3
 - a.5 参考文献 4
- b. API 设计 4
 - b.1 简介 4
 - b.2 API 内容（见最后附页） 4
- c. 界面设计 5
 - c.1 用户界面 5
- d. 业务跳转图 12
- f. 其它 14
 - 附录 A：词汇表 14
 - 附录 B：分析模型 14
 - 附录 C：待确定问题的列表 14

a. 引言

a.1 目的

说明书编写目的在于明确系统整体设计，界定系统实现功能的范围，指导系统设计以及编码。本说明书详尽说明了这一软件产品的设计思想，是整个软件开发的依据，它对以后阶段的工作起指导作用。本文也是项目完成后系统验收的依据。

a.2 修订历史

日期	版本	说明	作者
<2017/7/12>	V1.0	描述项目 API、界面、业务流程设计	路萍
<2017/7/12>	V2.0	勘误	路萍

a.3 预期的读者和阅读建议

本文档的主要内容共分 3 部分：API 设计、界面设计、业务跳转设计，另外还有 2 个附录。API 设计部分描述了本系统前后端之间交互的 API；界面设计部分对用户界面进行了描述；业务跳转设计部分对系统的业务流程进行了描述。本文档面向多种读者对象：

- （1）项目经理：项目经理可以根据该文档查看系统设计并不断修改。
- （2）设计员：对系统的设计录入此文档中。
- （3）程序员：配合《需求报告》，了解系统功能，编写《用户手册》。
- （4）测试员：根据本文档对软件产品进行更有针对性的功能性测试和非功能性测试。
- （5）推广人员：了解预期产品的设计思想。
- （6）用户：了解预期产品的功能。
- （7）其他人员：如部门领导、公司领导等可以据此了解产品的功能和性能。

在阅读本文档时，首先要了解产品的功能概貌，然后可以根据自身的需要对每一功能进行适当的了解。

a.4 产品的范围

本软件——SchooledIn APP，基于客户的需求，充分考虑了具体用户的实际情况。本产品主要适用于以下用户：

(1) 有意于在校内进行问答式的知识共享；

(2) 拥有 Android 手机的用户；

针对这些用户，可以完成以下业务：

(1) 在客户端上登录 SchooledIn APP；

(2) 可以实现以兴趣和问题驱动的知识咨询共享服务平台；

a.5 参考文献

无

b. API 设计

b.1 简介

项目中的 API 采用 REST 风格，面向资源，将前后端分离，使我们的后端实用性更强，服务范围更广。

b.2 API 内容（见最后附页）

b.2.1 RESTful_API

b.2.2 RESTful_API_answers

b.2.3 RESTful_API_comments

b.2.4 RESTful_API_essays

b.2.5 RESTful_API_questions

b.2.6 RESTful_API_studios

b.2.7 RESTful_API_users

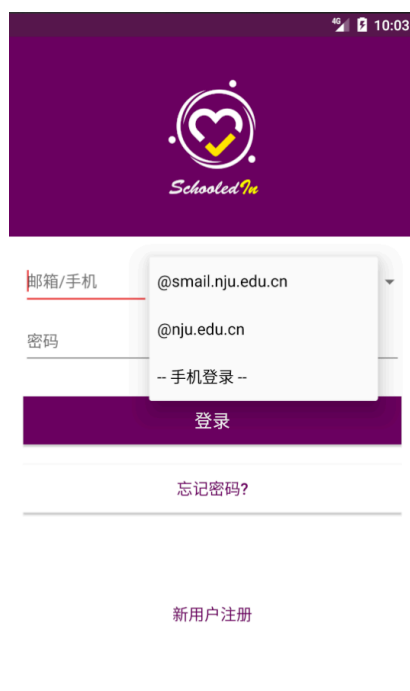
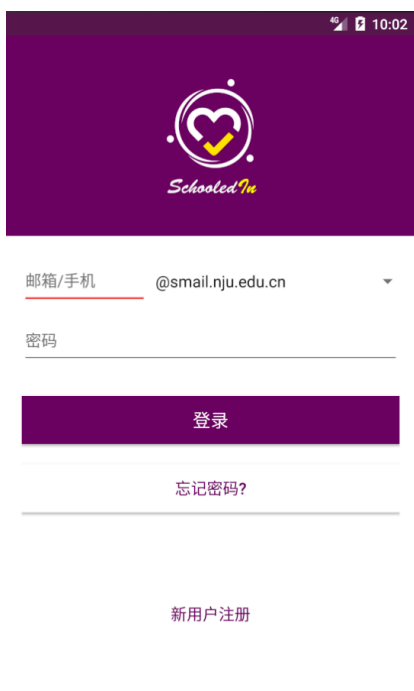
c. 界面设计

c.1 用户界面

1、启动



2、登录



3、注册



10:05

SchooledIn

请输入邮箱 发送验证码

请输入验证码

请输入密码

☐ 我已阅读并同意服务条款

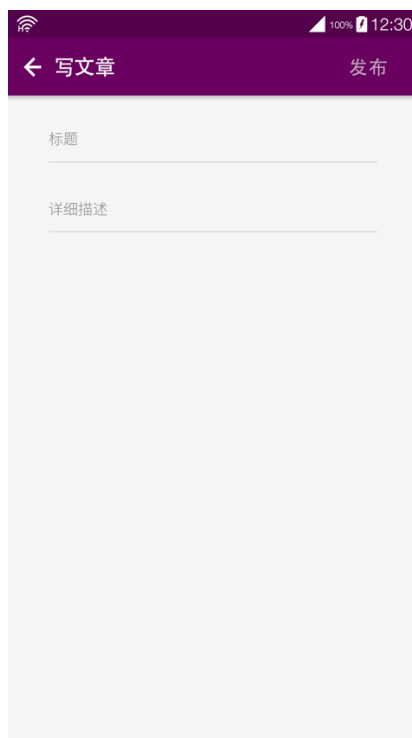
注册

我已经有账户了 登录

4、主页



5、发布文章



12:30

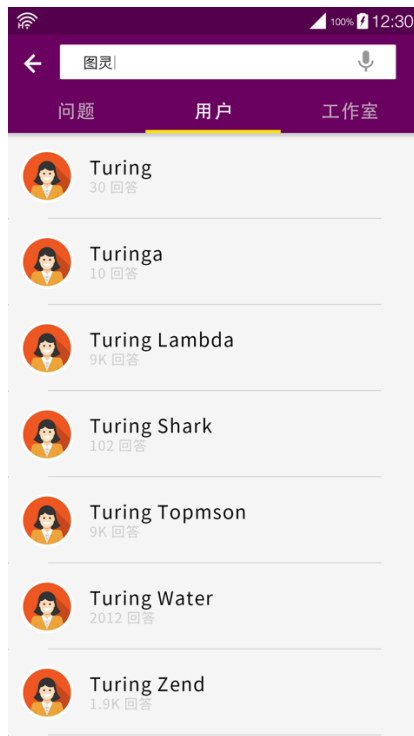
写文章 发布

标题

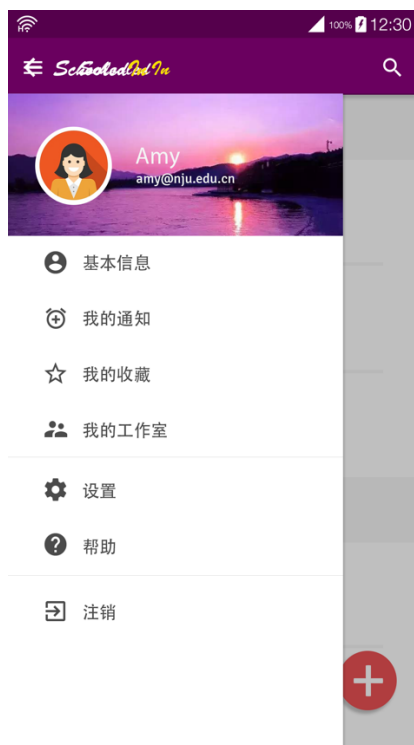
详细描述

6、搜索

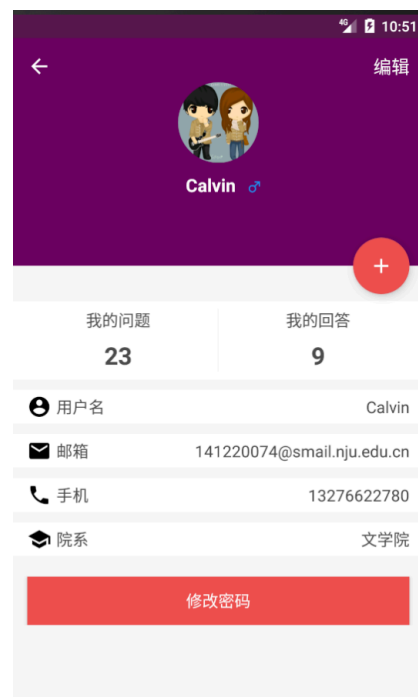




7、抽屉



8、个人信息





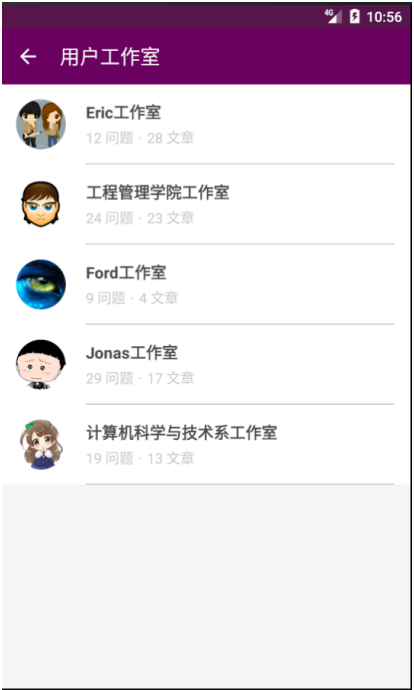
9、用户回答



10、用户问题



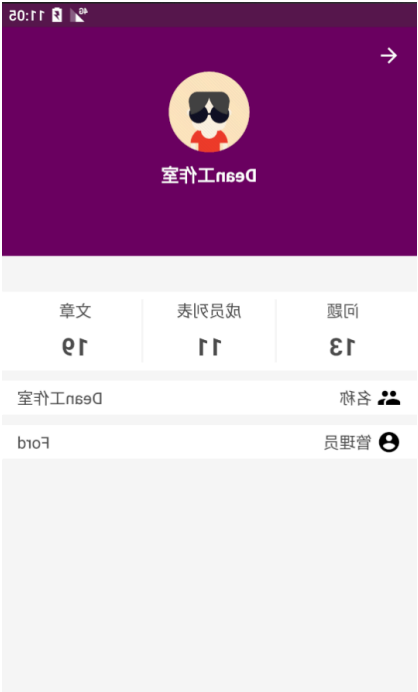
11、用户工作室



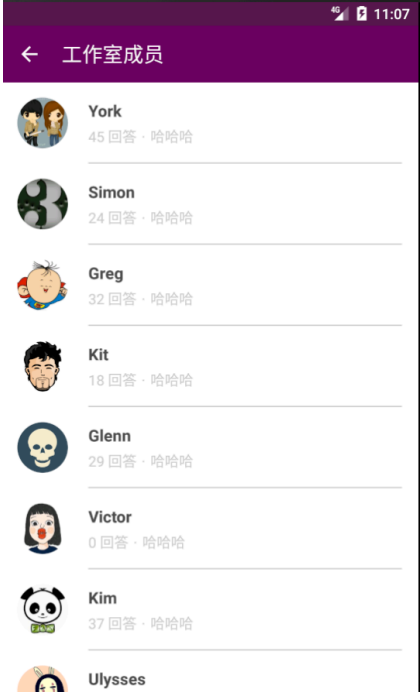
12、个人收藏



13、工作室信息



14、工作室成员列表



15、工作室回答过的问题



16、工作室文章



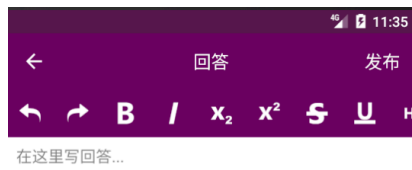
17、文章详情



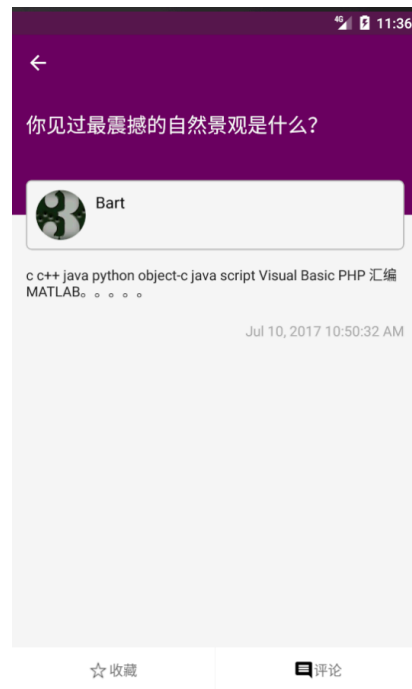
18、问题详情



19、发布答案



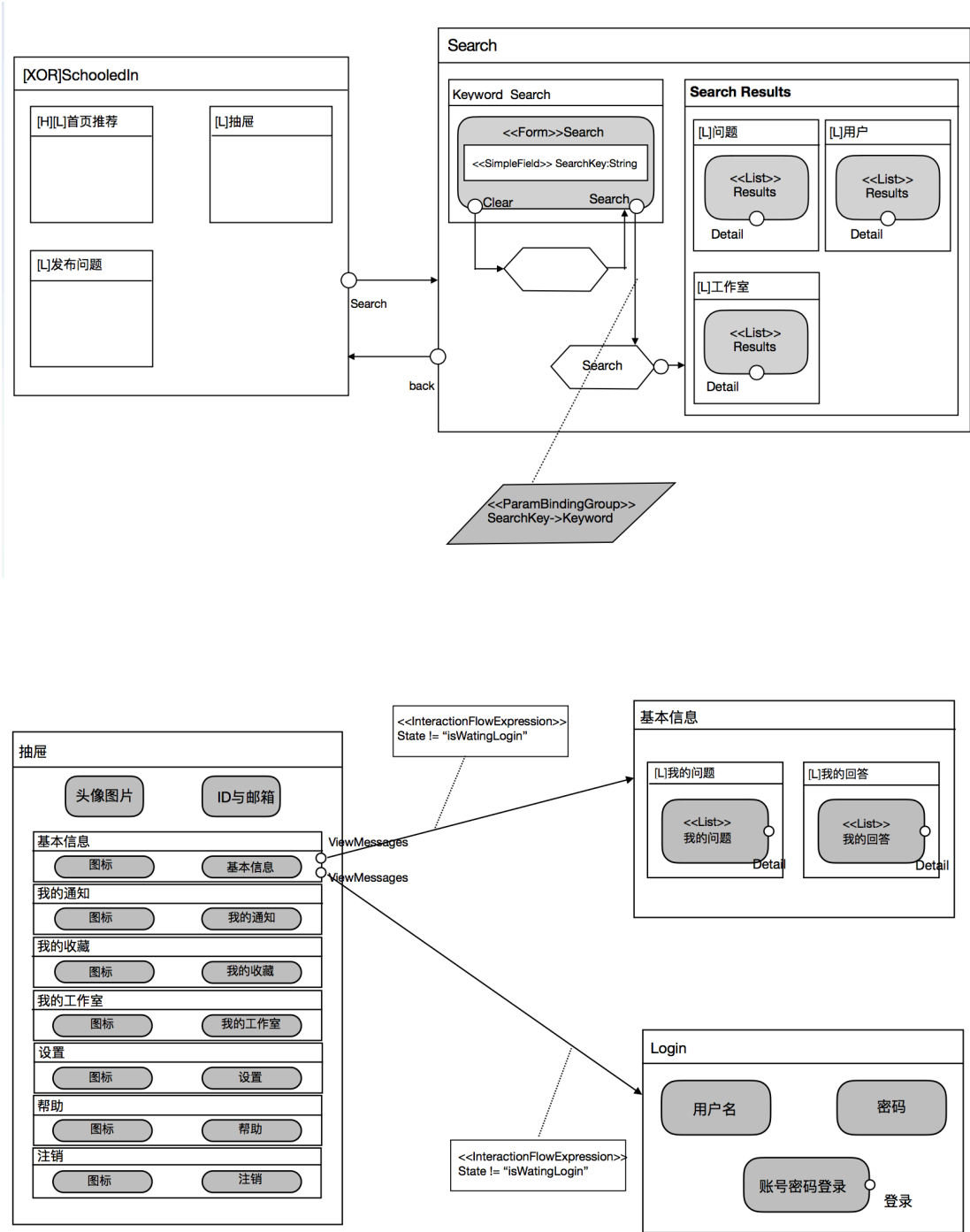
20、答案详情



21、评论详情

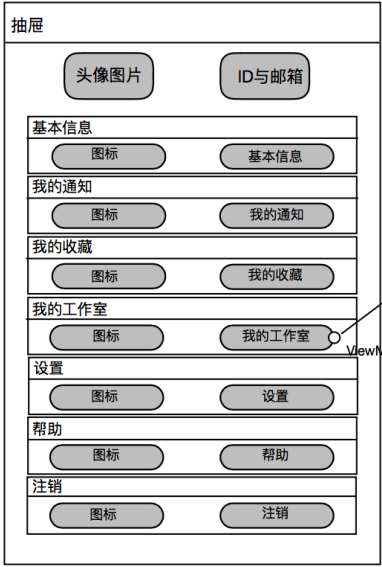
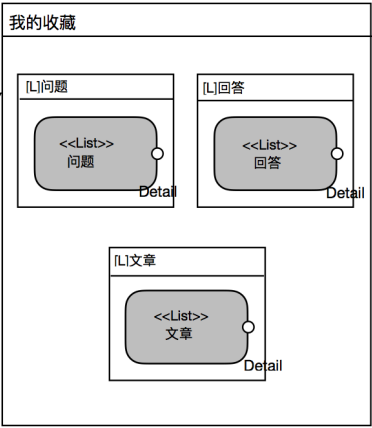


d. 业务跳转图

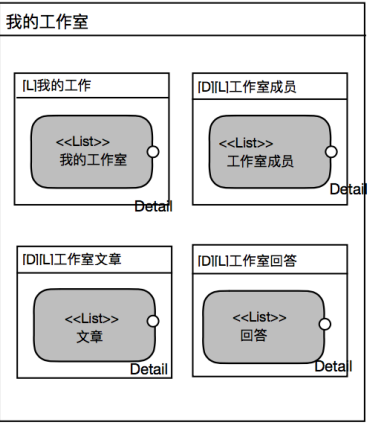




<<InteractionFlowExpression>>
State != "isWaitingLogin"



<<InteractionFlowExpression>>
State != "isWaitingLogin"



f. 其它

附录 A：词汇表

REST: 即表述性状态转移(Representational State Transfer, 简称 REST), 是一种面向资源的架构风格。它是一种针对网络应用的设计和开发方式, 可以降低开发的复杂性, 提高系统的可伸缩性。

RESTful: 一种软件架构风格, 设计风格而不是标准, 只是提供了一组设计原则和约束条件。它主要用于客户端和服务端交互类的软件。基于这个风格设计的软件可以更简洁, 更有层次, 更易于实现缓存等机制。

API : Application Programming Interface, 应用程序编程接口, 是一些预定义的函数, 目的是提供应用程序与开发人员基于某软件或硬件得以访问一组例程的能力, 而又无需访问源码, 或理解内部工作机制的细节。

附录 B：分析模型

无

附录 C：待确定问题的列表

无