

唠嗑婆婆小程序介绍文档

关注新闻时事，感悟生活百态

我非小白团队

2018.5

目录

1. 产品背景.....	3
1.1. 产品背景.....	3
1.2 产品概况.....	3
2. 需求分析.....	3
2.1 用户画像.....	3
2.2 需求汇总.....	3
3. 解决的实际问题.....	4
4. 前端界面详解.....	4
4.1 日话题.....	5
4.2 参与话题.....	6
4.3 周话题.....	7
4.4 排行榜.....	8
4.5 话题池.....	9
4.6 我的.....	10
5. 管理端界面展示.....	12
5.1 功能列表.....	12
5.2 话题池管理.....	13
5.3 话题管理.....	13
6. 功能讲解.....	14
6.1 功能框架图.....	14
6.2 用户使用路径.....	15
6.3 功能讲解.....	16
7. 技术分析.....	17
7.1 前端技术分析.....	17
7.2 后端技术分析.....	21
7.3 管理端.....	23
8. 市场分析.....	26
9. 团队介绍.....	28
10. 总结.....	29

1. 产品概述

1.1. 产品背景

微信小程序的优势在于“用完即走”，我们以用户能够即时跳出小程序查看微信消息且再次进入小程序不受影响为目标，同时考虑到生活中人们对于新闻时事的关注以及碰到烦恼想要倾诉解惑的需求，设计出了唠嗑婆婆这款小程序。希望人们在使用唠嗑婆婆时能够倾诉自己的困扰，倾听他人的想法，形成良好的话题社区环境。

1.2 产品概况

“关注新闻时事，感悟生活百态”，唠嗑婆婆是一款话题讨论小程序，用户每天可以参与系统围绕新闻时事热点发出的日话题进行讨论。每周全国排行榜前五名的用户将会获得向话题池中添加话题的权利，同时话题池中获赞最高的话题将会作为下周周话题。

2. 需求分析

2.1 用户画像



2.2 需求汇总

序号	应用场景	问题需求	对应功能
1	即将毕业，找工作不顺利，对	想听听其他人的	发布话题，让其他

	生活或很迷茫	经历和意见	人参与
2	看到很好的话题回答很想收藏分享	能够收藏话题以及喜欢的话题回答并分享给朋友	按照话题列表将用户收藏的话题自动分类,同时能够一键分享
3	其他平台上发布的话题没人关注	想要将自己的话题成为热门,获得他人关注和参与	每周根据排行榜情况获得发布话题的权利,得赞最多的成为下周话题
4	想要知道某个好友对于这个话题的看法,又不想直接询问	能够知晓好友对于某个话题的看法	每当有好友回答相关话题时就及时推送给用户,用户可以点击查看
5	新闻时事很难清楚的看到别人的看法	不要出现评论回复,直接看到观点就行	不提供回复评论的功能,没有楼中楼的困扰,避免出现争吵现象。

3. 解决的实际问题

1: 倾吐生活中的烦扰困惑

人生中的不同时期总会遇到不同的困惑,这些困惑或许来自于生活,或许来自于工作,又或许来自于自我的成长思考。面对这些困惑,你可能很想找他人倾诉,听听他人的想法。而有的时候一些好的建议可能会改变

一个人的未来，成为人生道路的启明灯。而唠嗑婆婆正是一个可以让用户倾吐心声，寻找答案的平台。

2: 有趣有意利用碎片时间

用户可以利用碎片时间在唠嗑婆婆中查看话题回答，针对一些优质话题可以进行自我思考，同时也可以看到一些优质回答来帮助从另一方面思考这个问题，帮助自己看待问题的时候更加全面，获得自我的提升。将碎片的时间利用起来，关注新闻时事，了解社会最新动态，同时感悟生活。

4. 前端界面详解

4.1 日话题



➤ 历史话题

1. 点击日期，将会向下弹出往期每天的话题列表，点击话题即可查看当

天话题的回复。

2. 点击最下方的日期即可出现多选框，选择需要跳转的月份

➤ 日话题页面

1. 中间的图和话题内容以及字的颜色由后台人员每天制作好上传即可

2. 下方为用户话题答复区域，用户可以选择按热度或按时间排序。对于用户喜欢的话题可以点击每个话题下方的按钮进行收藏和点赞

4.2 参与话题



➤ 参与话题

1. 用户可以在输入框内输入文字，字符总数不得多余 500

2. 点击照片按钮，用户可以选择拍照或者从手机相册中选择一张照片上传

3. 编辑完毕后可以点击发表按钮进行发表

4.3 周话题



➤ 周话题

1. 点击日期，跳出话题列表，话题按照周排序，七天一个间隔
2. 点击加号，跳到参与话题页面，回答话题

4.4 排行榜



➤ 排行榜

1. 昵称下面为日话题点赞总数和周话题点赞总数相加，最后的数字为相加的和。
2. 最下面为用户自己在全国排行榜的位置，能够查看到当前自己所处的位置和自己被点赞的总数
3. 用户可以在好友榜里查看自己在好友榜中的排名，同时查看好友排名以及好友被点赞总数

4.5 话题池



➤ 话题池

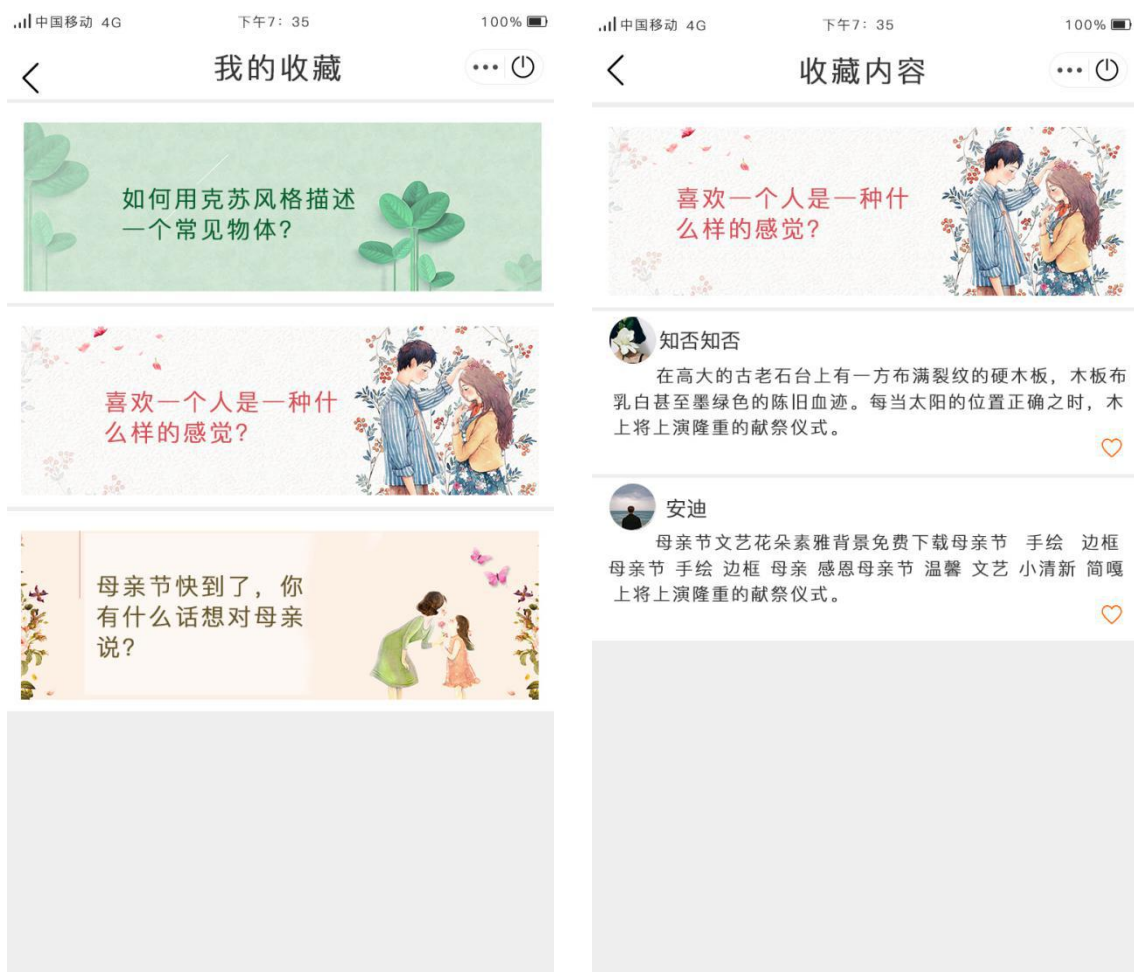
1. 话题池里为特殊用户添加的话题，其他用户可以在话题池里查看话题，可以为自己喜欢的话题点赞。
2. 对于喜欢话题可以点击分享按钮，分享到朋友圈让好友点赞

4.6 我的



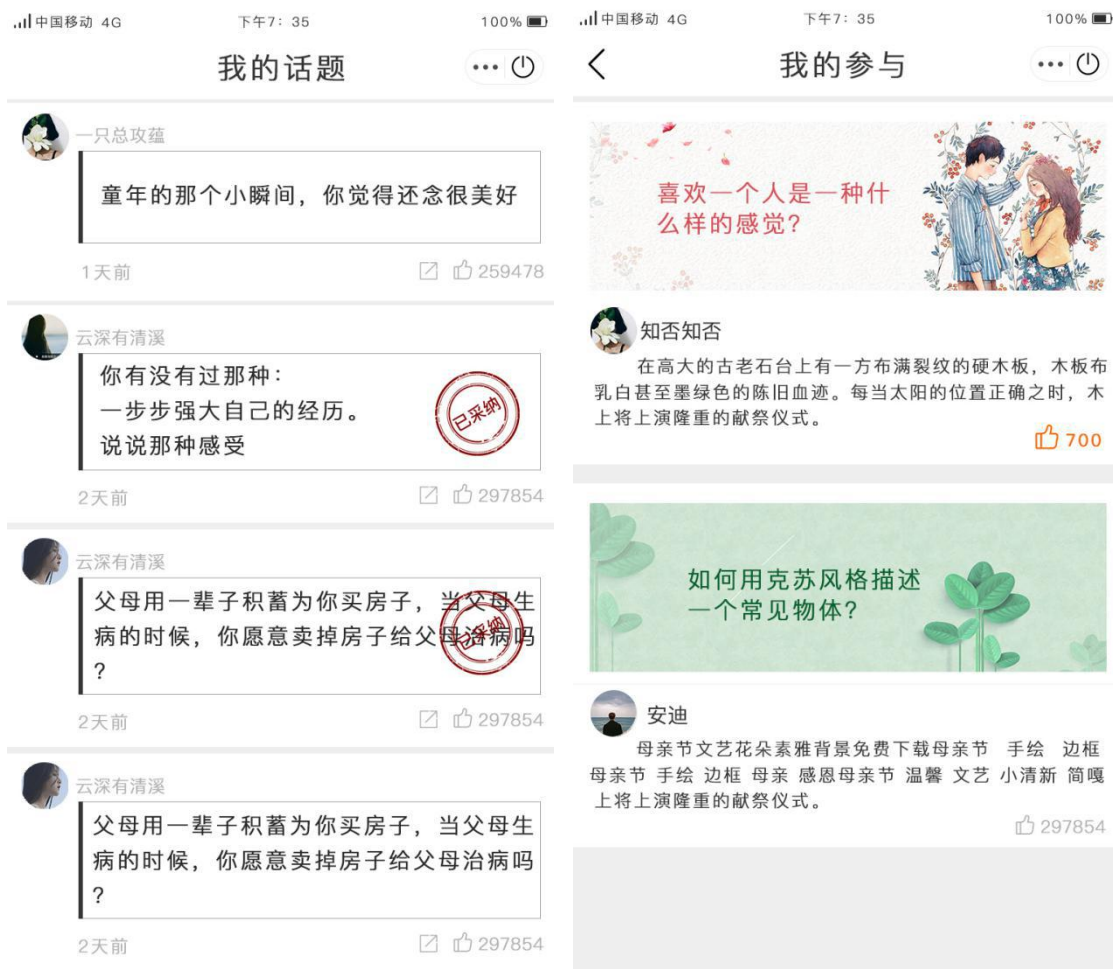
➤ 消息

1. 赞为他人给用户点赞时候的提示，用户可以在此查看是谁给自己哪条话题点赞
2. 下方为好友参与话题时的提示，用户可以点击查看好友关于某个话题的答复



➤ 我的收藏

我的收藏为用户收藏的话题回复，点击以后为话题列表展示，只有被收藏过回复的话题才会显示在里面，用户点击列表即可查看自己收藏的话题。同时点击收藏按钮即可取消这个话题。



➤ 我的话题

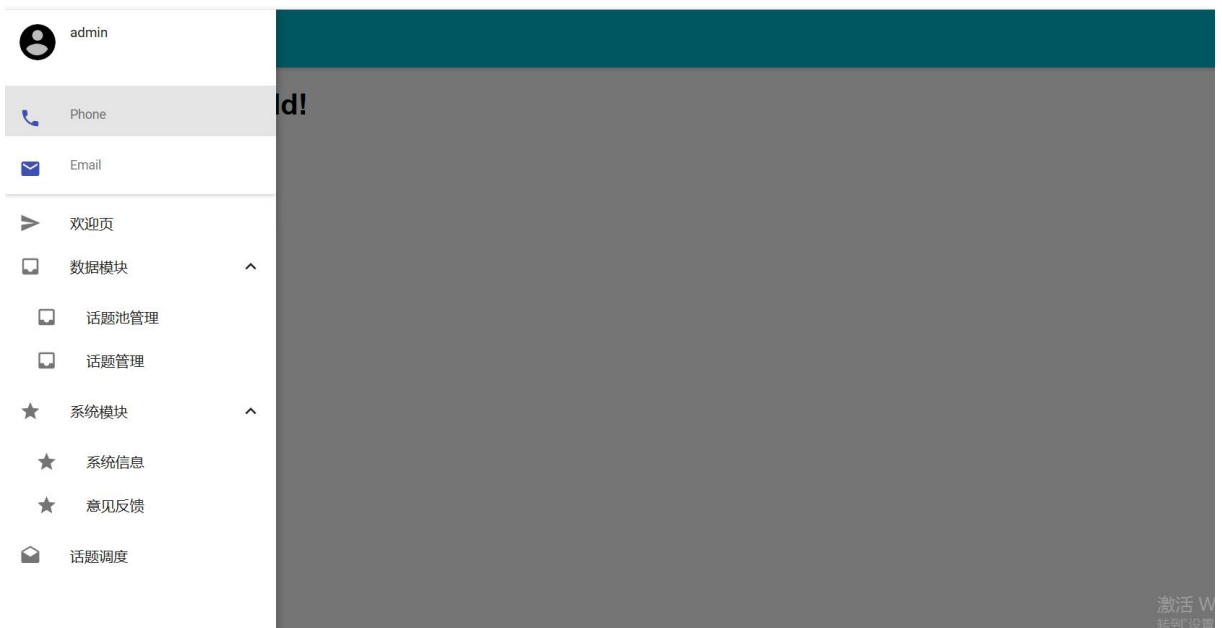
我的话题为用户获得机会后往话题池中添加的话题，如果某个话题获赞最高，被作为周话题，则这个话题将会被标注为已采纳，用户可以通过这边直接跳进到周话题页面查看这个话题的答复。

➤ 我的参与

我的参与可以查看我关于某个话题的答复，对于今天和本周的话题我的答复可以进行修改，而以前的则不可以修改。上方为话题内容，下方为我关于这个话题的回答。

5. 管理端界面展示 (<https://d0vnlgdt.qcloud.la> 密码：123456)

5.1 功能列表



5.2 话题池管理



5.3 话题管理

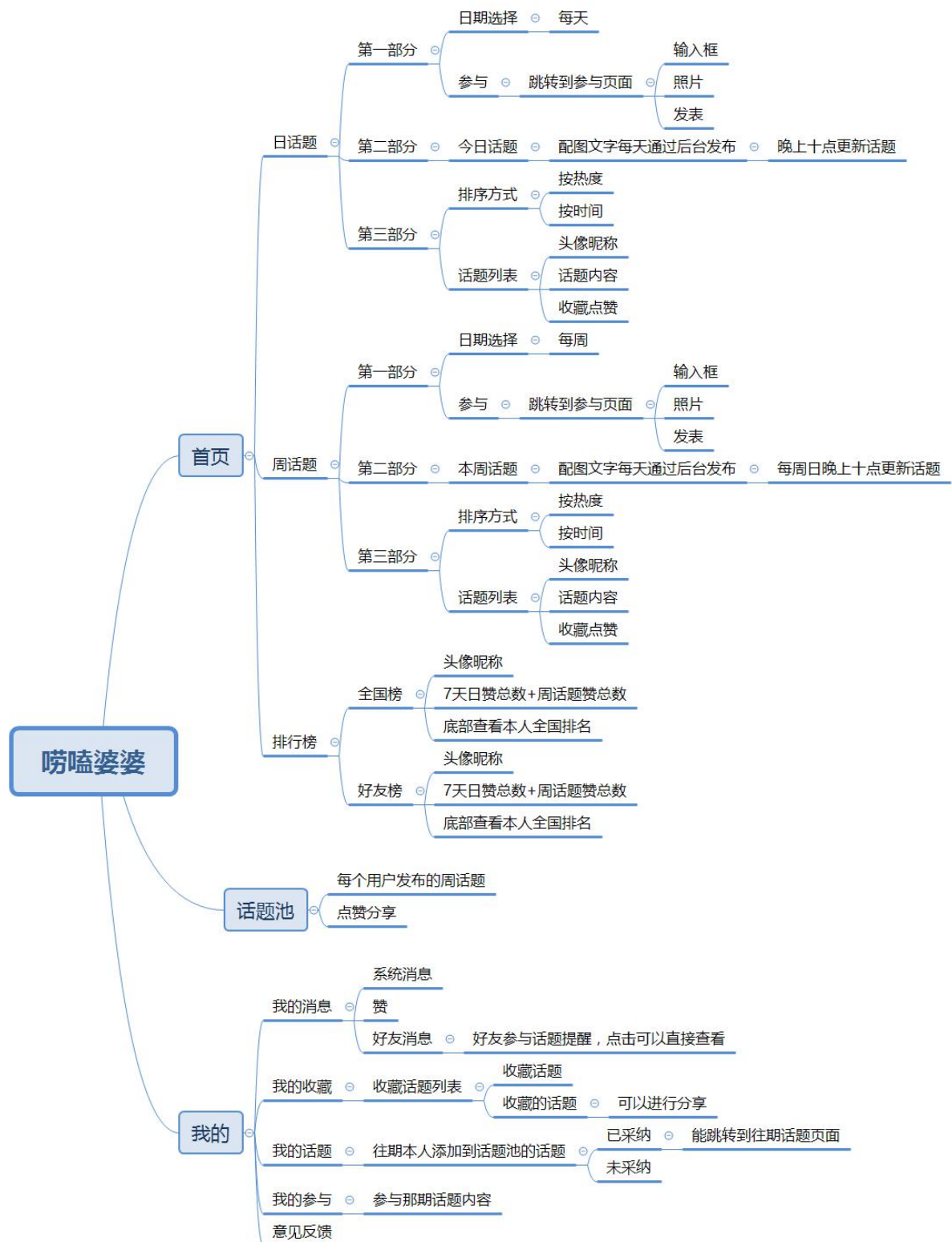
≡ 管理系统

退出登录

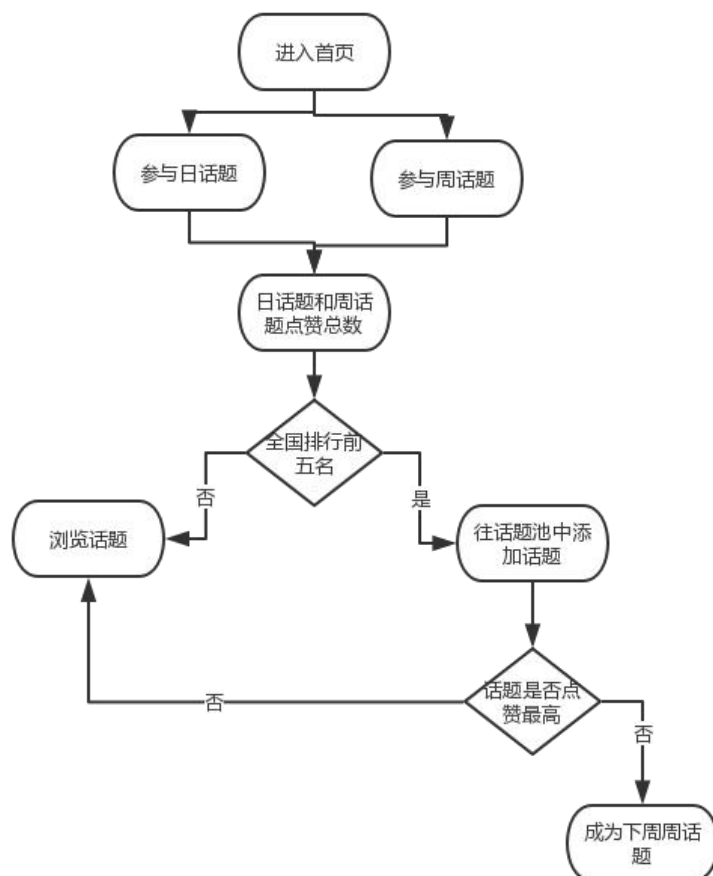
每页条数 10	页码 1	GO	话题	修改	删除	刷新
<input type="checkbox"/>	ID	类型	Pool ID	时间		
<input type="checkbox"/>	29	日话题	15	2018-06-06T19:17:28.000Z		
<input type="checkbox"/>	28	周话题	16	2018-06-06T19:15:42.000Z		
<input type="checkbox"/>	27	日话题	13	2018-06-06T19:15:35.000Z		
<input type="checkbox"/>	26	日话题	15	2018-06-06T19:12:56.000Z		
<input type="checkbox"/>	25	日话题	12	2018-06-06T19:11:51.000Z		
<input type="checkbox"/>	24	周话题	11	2018-06-06T18:26:37.000Z		
<input type="checkbox"/>	23	周话题	15	2018-06-06T18:12:41.000Z		
<input type="checkbox"/>	22	周话题	16	2018-06-06T18:09:41.000Z		
<input type="checkbox"/>	20	日话题	16	2018-06-06T18:03:47.000Z		
<input type="checkbox"/>	19	日话题	10	2018-06-04T14:25:32.000Z		

6. 功能分析

6.1 功能框架图



6.2 用户使用路径



6.3 功能讲解

6.3.1 日话题功能

1. 每天晚上 10 点上新新话题，同时统计当天的每个话题点赞总数，刷新当天的排行榜

2. 话题区每个用户只可以发布一条话题答复，可以针对这回答进行多次修改，修改以后将会作为最新发布的答复。

3. 小程序不存在楼中楼的情况，用户不可以进行评论和回复。同时用户离开小程序时我们将会记录用户所看话题的位置，用户下次进来时可以回到刚才所看话题的位置。

6.3.2 周话题功能

1. 每周日晚上 10 点刷新周话题排行榜，话题来源于话题池中点赞最高

的话题。

2. 话题池中的话题来源于全国排行榜前五名用户添加的话题。获得权利的用户可以在周日晚上 10 点以后往话题池中添加话题，每个用户只有一次添加的权利。

6.3.3 收藏分享

用户可以将收藏的话题通过页面的方式分享给朋友或者朋友圈，其他用户可以点击进入小程序查看用户收藏的话题。

6.3.4 好友通知

当好友参与了某个话题的讨论，用户将会收到消息提示，同时在消息中可以直接查看好友关于某个话题的回答。同时用户可以在好友榜中查看好友排行，同时能够看到自己在好友中的排行。

7. 技术分析

7.1 前端技术分析

7.1.1 简述

(1) 开发平台：微信开发者工具 V1.02.1803210

(2) 开发框架：wxapp-devFrame

7.1.2 优势

(1) 使用小程序基本的开发框架 wxapp-devFrame，对网络访问进行了二次封装，只需要配置页面 url，请求参数，返回的数据每个 item 字段的处理，wxml 里写 item 的布局即可。无需关心页面状态，刷新和加载更多的 ui 和数据拼接。

(2) WXML 支持数据绑定、逻辑算术运算、支持模板及引用、添加事件

(3) WXSS 支持大部分 CSS 特性，添加尺寸单位 rpx，可根据屏幕宽度

自适应

(4) 不支持多层选择器，避免被组件内结构破坏

(5) 小程序的组件基于 Web Component 标准，使用 Polymer 框架实现 Web Component

(6) View 层和逻辑层分离，通过数据驱动，不直接操作 DOM

(7) 项目采用 MVC 模式

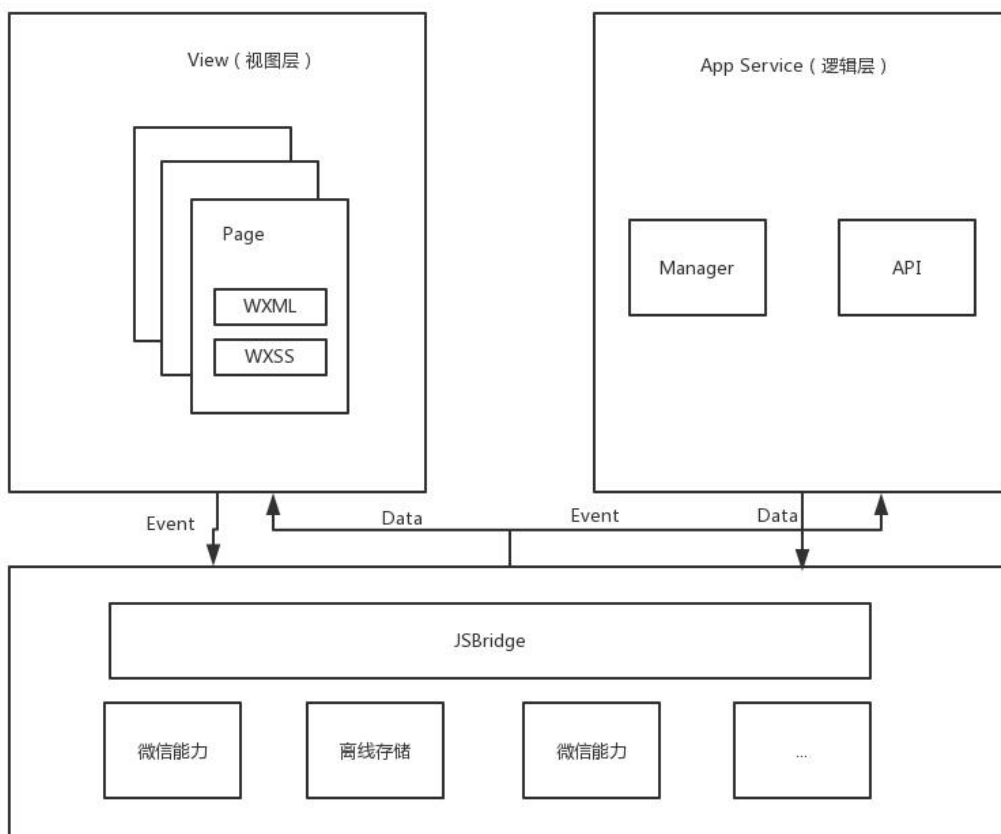
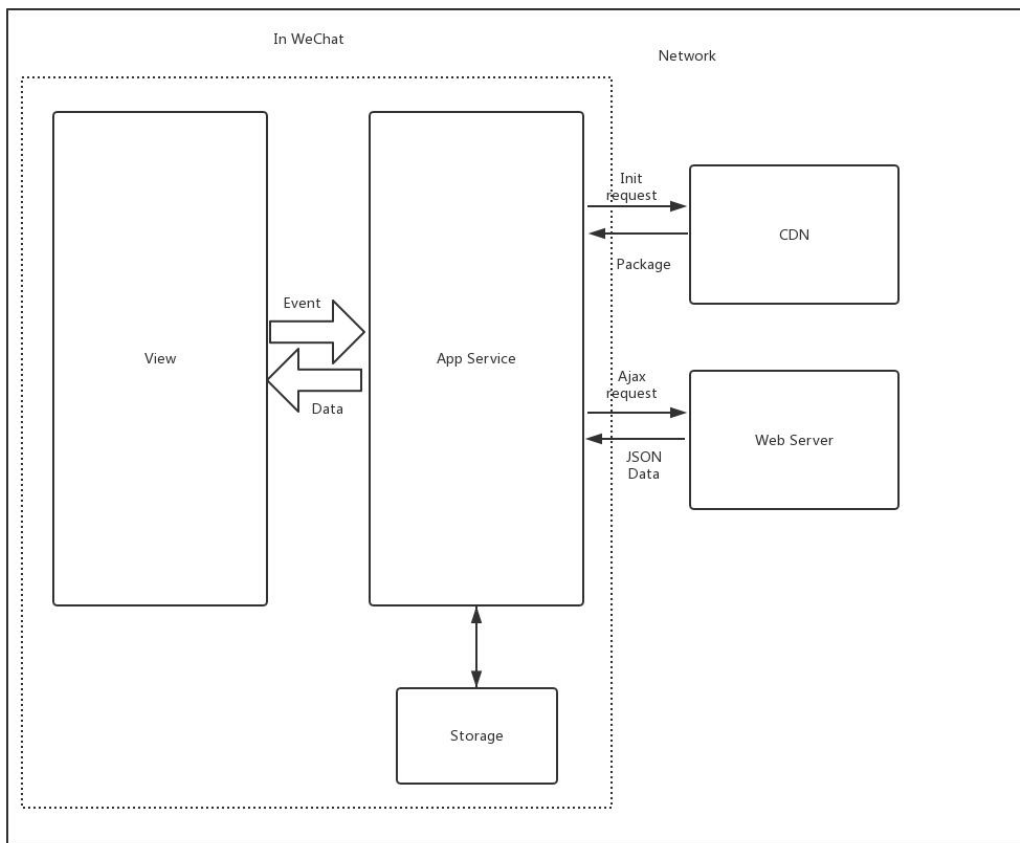
● 大部分用过程语言比如 ASP、PHP 开发出来的 Web 应用，初始的开发模板就是混合层的数据编程。例如，直接向数据库发送请求并用 HTML 显示，开发速度往往比较快，但由于数据页面的分离不是很直接，因而很难体现出业务模型的样子或者模型的重用性。产品设计弹性力度很小，很难满足用户的变化性需求。MVC 要求对应用分层，虽然要花费额外的工作，但产品的结构清晰，产品的应用通过模型可以得到更好地体现。

● 由于一个应用被分离为三层，因此有时改变其中的一层就能满足应用的改变。一个应用的业务流程或者业务规则的改变只需改动 MVC 的模型层

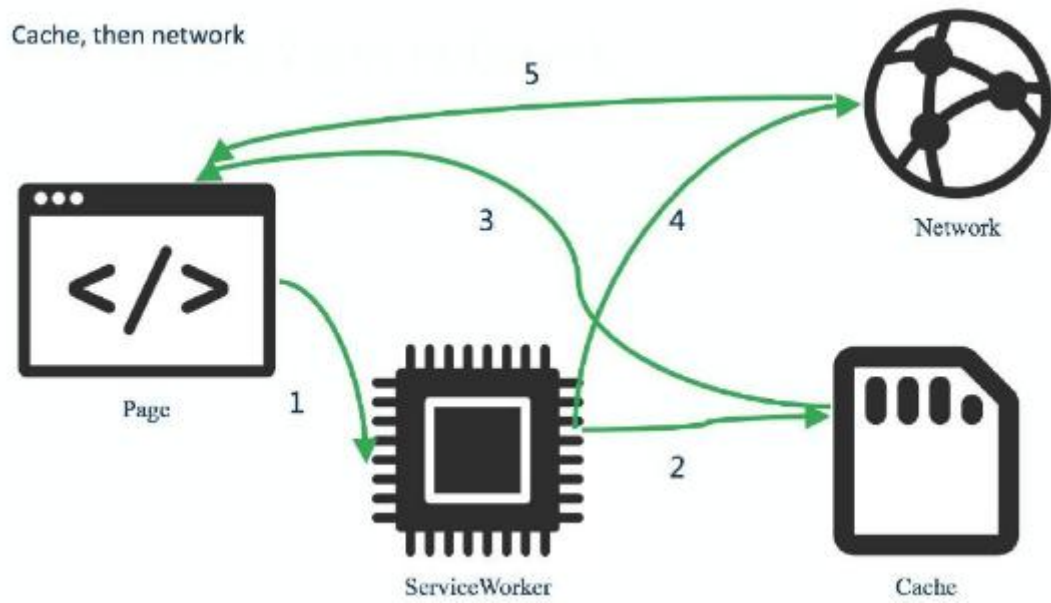
● 控制层的概念也很有效，由于它把不同的模型和不同的视图组合在一起完成不同的请求，因此，控制层可以说是包含了用户请求权限的概念。

● 有利于软件工程化管理。由于不同的层各司其职，每一层不同的应用具有某些相同的特征，有利于通过工程化、工具化产生管理程序代码。

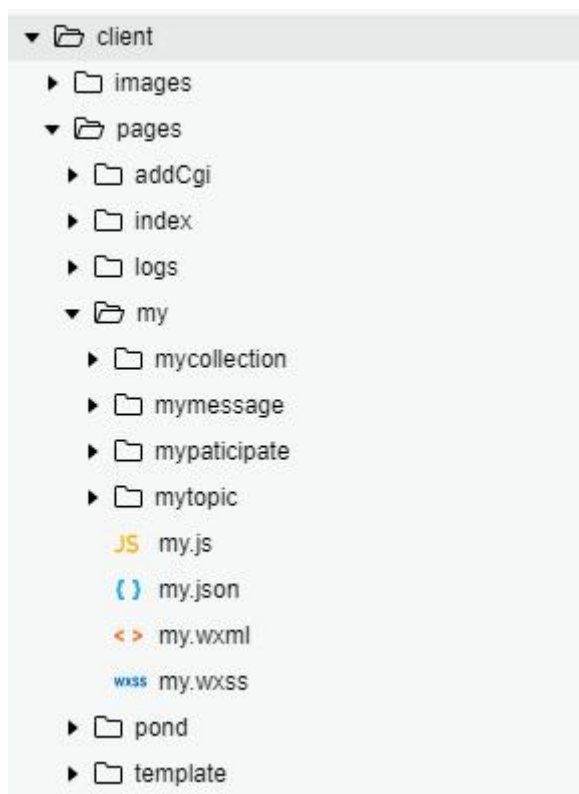
7.1.3 小程序架构

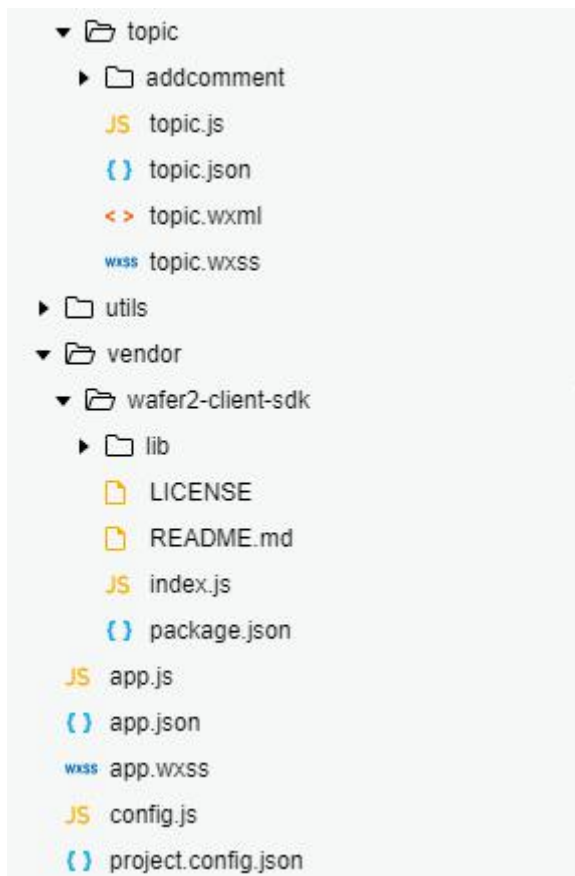


通过 Service Works 实现资源离线缓存和更新



7.1.4 项目架构





7.2 后端技术分析

7.2.1 简述

- (1) 开发平台&运行平台：nodejs
- (2) SDK：微信小程序 node sdk
- (3) 数据库：mysql
- (4) web 开发框架：koa
- (5) http 请求框架：axios
- (6) 数据库框架：knex

7.2.2 package.json 节选

json

{

... ,

```

"dependencies": {
  "axios": "^0.15.3",
  "knex": "^0.13.0",
  "koa": "^2.0.0",
  "koa-bodyparser": "^3.2.0",
  "koa-logger": "^2.1.0",
  "koa-router": "^7.0.1",
  "koa-static": "^4.0.3",
  "lodash": "^4.17.4",
  "mkdir-p": "0.0.7",
  "mysql": "^2.14.1",
  "pify": "^2.3.0",
  "wafer-node-sdk": "^1.3.5"
},
"devDependencies": {
  "babel-eslint": "^7.1.0",
  "debug": "^2.6.8",
  "eslint": "^3.9.1",
  "eslint-config-standard": "^6.2.1",
  "eslint-plugin-promise": "^3.3.1",
  "eslint-plugin-standard": "^2.0.1"
}
}

```

7.2.3 优势

- (1) 使用 RESTful 架构对外提供接口，保证了后端接口风格的统一
- (2) 使用 koa & knex 保证了后端的稳定运行
- (3) 开发遵循 MVC 结构，便于各模块解耦

7.3 管理端

7.3.1 简述

- (1) 开发平台&运行平台：nodejs
- (2) 前端框架：React + React Router + React Redux
- (3) 打包工具：Webpack
- (4) UI 框架：Material-UI
- (5) http 请求框架：axios

7.3.2 package.json 节选

```
```json
{
 ...,
 "dependencies": {
 "axios": "^0.18.0",
 "material-ui": "^0.20.1",
 "react": "^16.4.0",
 "react-dom": "^16.4.0",
 "react-redux": "^5.0.7",
 "react-router-dom": "^4.2.2",
 "react-scripts": "1.1.4",
```

```
 "redux": "^4.0.0"

 },

 ...

}

...
```

### 7.3.3 优势

- (1) 单页面多路由，加快管理端前端 Web 页面的性能
- (2) 使用 Material-UI 使页面符合 Material Design
- (3) 开发遵循 MVC 结构，便于各模块解耦

### 7.3.4 界面概览

#### 1. 入口

![login](\_v\_images/\_login\_1528390729\_21725.png)

#### 2. 菜单

![菜单](\_v\_images/\_1528390773\_10086.png)

#### 3. 话题池管理



#### 4. 话题管理



#### 5. 分页



#### 6. 意见反馈







## 7. 其它



### 7.3.5 详细说明

#### 1. 项目整体目录结构



- + components: 通用组件
- + reducers: 使用 Redux 进行页面状态管理
- + routes: 页面路由
- + error: 错误页 (400、500)
- + styles: 全局样式文件
- + utils: 工具类
- + index.js: 程序入口
- + XXXX.jsx: 基本页面布局
- + XXXXContainer.js: 组件容器、给页面提供 Redux 的 connect、

React-router 的 withroute 等包装

#### 2. MVC 说明

- + XXXX.jsx 负责页面的展示以及简单的页面逻辑
- + XXXXContainer.js 作为页面的控制器
- + xxxReducer.js 处理数据，修改全局状态，通知 jsx 重新渲染页面

## 8. 市场分析

### 8.1 微博搞笑排行榜数据分析



微博搞笑排行榜是每天会发布一个关于生活方面的话题，让粉丝在下方评论区讨论。目前粉丝人数为 4141 万，由此可见每天一个话题的模式有很多人能够接受。同时每天话题讨论人数达到 203615（截取其中一个），同时转发和点赞人数也尚可。这说明关于话题有很多人关注喜欢。榜姐每天凌晨 12 点更新话题，有忠于榜姐的粉丝会在 12 准时抢沙发，这也形成了一种娱乐模式。所以由此看来唠嗑婆婆的模式可行。

## 8.2 微博话题热门榜



微博话题榜采取的是发布一个导语，用户通过发布微博加上话题名称即可参与话题讨论。由图中可以看出热门话题有 513.4 万讨论，36.3 亿阅读。一个很好的话题能引起很多人的共鸣并参与讨论。话题的用户人群及市场拥有很大规模。

### 8.3 知乎话题



知乎中也有话题榜这个概念，如图中可以看出，2018 高考话题的关注度达到 1095 万热度，点开以后可以看到有 3460 个回答，而回答下方点赞次数达到 6146. 如此可以看出，话题这个概念是很有关注度的。

### 9. 团队介绍

责任小组	责任	责任人
项目经理	负责项目的推进 负责小组任务的下达与跟踪 审批顾问组提交的项目实施进度表 设定项目目标，范围及评价考核标准	韩颖
顾问组	负责整个软件开发项目的规划设计 负责项目组织架构建立与调整	全体人员

	负责流程制定、修改与维护	
需求分析师	负责项目相关需求分析与获取，提交需求分析文档。并对需求分析结果负责	韩颖
网络架构师	网络整体架构的规划与设计 重点网络项目的设计、实施及项目管理 核心应用网络方案的设计与实施	莫音子
界面/交互设计师	根据需求和设计要求提供用户使用分析，使用情景流程设计用户界面和交互设计，为用户界面和交互设计实施提供设计标准规格，详细说明书	韩颖
Web 前端/后端管理	负责管理端和数据库设计	莫音子
前端开发	负责客户端的代码具体实现	朱熠雯
测试人员	对需求的把握、协助保证前期的规划，并对整个软件开发的过程进行监督； 根据需求分析，进行功能测试和性能测试。检查是否符合需求和是否存在 BUG，以及提高软件的可操作性和界面的友好性。	全体人员

## 10. 总结

唠嗑婆婆通过话题参与的方式让人们关注时事生活，通过点赞量排名的方式获取发布话题的机会，用户选择出最喜欢的话题作为下周话题，从而形成公平竞争。在学习中玩，不失为一种乐趣。取消楼中楼的形式，避免出现争吵的现象，从而形成良好的社区氛围，也让用户在浏览的时候更加清溪明了。增加好友消息通知，促进好友之间的交流，增进人与人之间的情感。

在短短几个月里面，我们通过网上问卷调查，了解用户的需求，听取他们的意见。各方面的了解分析和设计，做出了如今的唠嗑婆婆产品。技术上，我们讨论研究，百度学习新的知识，一个一个的攻破难题。不让技

术成为产品的后腿。如果大家对于我们的“唠嗑婆婆”有更好的提议和方案，欢迎一起交流、切磋。有了更多的用户体验我们才能做出更好的产品。

谢

