

## 基于 Web 平台家长帮社区 APP 设计与实现

**[摘要]**随着中国教育的进步,家长越来越重视对孩子的教育,根据国内的形势,无论一个家庭环境是富裕还是贫穷,家长都力求通过自身能力让孩子身心健康得到全方位的发展。但存在家长因自身认知,客观看法形成不好教育思维,导致了孩子的成长道路出现与预期的偏差。

移动智能手机的普及,App 即移动应用程序,已经作为不可忽略的数字媒介借助网络时代的发展踏入家庭教育领域。打造以孩子为中心话题的家长群体社区应用,形成一个家长间互相交流互动的沟通平台,正是我们当下家庭教育所需。如何通过社区互动解决家长在养儿,育儿,教儿中资讯不平衡,方法不对,改变家长自身教导的错误认知,借鉴优秀教育方法,形成有效的群体交流正是本论文应用的主要目的。

本应用通过借鉴国内现有的家长社区应用,取其精华,分析传统家庭教育的不足,总结家长在孩子教育中的所遇到的问题,对基于 Web 平台的家长帮社区 APP 进行国内背景的调研,分析了应用的市场定位,对应用的需求分析,所应用的技术工具,测试过程,设计与实现过程进行详细的阐述,本论文最终测试结果良好,能够达到预期效果。本论文不仅论述如何设计并实现一个基于 Web 平台的 APP,也可以作为 Web APP 项目开发的最佳实践借鉴。

**[关键词]**Web; 社区; 教育; 家长

## **The design and implementation of Parents Help Community APP based on Web**

**[Abstract]**With the progress of education in China, parents pay more and more attention to the education of their children. According to the domestic situation, regardless of whether a family environment is prosperous or poor, parents strive to achieve all-round development of their physical and mental health through their own abilities. However, due to the fact that parents have formed poor educational thinking because of their own perceptions and objective views, they have led to deviations from the expectations of children's growth.

With the popularity of mobile smart phones, App, as a mobile application, has entered the family education field with the development of the Internet era as a non-negligible digital medium. To create a parent-group community application that focuses on children and form a communication platform for parents to interact and interact with each other is exactly what we need for current family education. How to solve parental problems in raising children through child-rearing, child-rearing, and teaching is not correct. It is not correct to change parental self-education, to learn from excellent educational methods, and to form effective group communication is the main purpose of this paper.

This application draws on domestic existing parent community applications, takes its essence, analyzes the shortcomings of traditional family education, sums up the problems parents face in their children's education, and conducts research on the background of the parent-assisted community APP based on the Web platform. The paper analyzes the market positioning of the application, analyzes the requirements of the application, the applied technical tools, the testing process, and the design and implementation process. The final test results are good and meet the expected requirements. This question not only discusses how to design and implement an application based on the Web platform, but also can be used as a reference for the development of Web APP projects.

**[Key words]**Web; Community; Education; Parents

# 目 录

基于 Web 平台家长帮社区 APP 设计与实现 .....	I
第 1 章 引言 .....	1
1.1 研究背景和意义 .....	1
1.2 家长教育现状和发展趋势 .....	1
1.3 本文的论文结构和章节安排 .....	2
第 2 章 关键技术与工具 .....	3
2.1 关键技术 .....	3
2.2 相关工具 .....	4
第 3 章 系统分析 .....	6
3.1 研究目的 .....	6
3.2 可行性分析 .....	6
3.3 需求分析 .....	7
3.4 系统开发环境 .....	13
第 4 章 系统的总体设计 .....	14
4.1 系统架构设计 .....	14
4.2 系统功能架构图 .....	15
4.3 系统各模块流程图 .....	15
第 5 章 系统详细设计 .....	21
5.1 系统模块介绍 .....	21
5.2 系统数据库设计 .....	24
第 6 章 系统的实现 .....	27

6.1 开发环境搭建 .....	27
6.2 安装关键依赖库 .....	30
6.3 工程结构说明 .....	31
6.4 系统页面效果 .....	32
6.5 部分功能代码分析.....	38
第 7 章 系统测试 .....	42
7.1 单元测试 .....	42
7.2 功能测试 .....	43
第 8 章 总结与展望 .....	46
8.1 工作总结 .....	46
8.2 不足与展望 .....	46
参考文献 .....	47
致 谢 .....	48

# 第 1 章 引言

## 1.1 研究背景和意义

随着经济的发展以及人文的进步,家长越来越重视对孩子的培育,力求孩子能够全方位的发展。家长往往是孩子的第一教师,他们的教育素养可以直接影响到教育的质量以及孩子的身心健康发展。

传统的家庭教育观念容易局限于家长的个人认知水平和行为习惯,不能够深层地解孩子的想法,容易造成无效的沟通;在学校与家长的关系上,老师很难顾及到每一位学生,如有问题需要请教,则需要打电话沟通或面对面交流,沟通效率较低下等。

而随着现在信息化的快速发展,在互联网时代,我们需要更多有效便捷的沟通,及时了解孩子的情况,得到有效的沟通,而现在的信息技术的进步,完全可以为我们提供一个高效的平台。结合我国社会教育形势,本选题应用提供一个让家长能够分享资讯,资源共享的应用,针对家长对孩子的培育建立信息与资源共享平台,致力于家长打造最实用的信息服务与最便利的交流社区,分享多元化的教育方式。

本课题不仅弥补了绝大部分家长难以获取可靠信息的现象,从而大大提高家长的教育质量,让家长更加了解孩子的心理需求,消除因不理解,代沟造成的错误教导,同时本应用提供对孩子进行成长记录的功能,使家长专注与孩子的成长,满足了家长社交的需求,适应了当代时代咨询互动的发展,具有时代创新性。

## 1.2 家长教育现状和发展趋势

家长教育,是指家长基于自身经历或者孩子的不同成长历程逐渐形成的教导认知,这种教导认知直接形成家长对孩子教育的手段,方法,目标以及方向。中国如今已经进入现代化时期,随着家长自身教育水平的提升,家长的教育观念有了明显的进步,慢慢摆脱了传统教育观念的束缚,但仍有稂莠不齐的现象。家长容易受到自身个体的客观思维影响,在教育上面误导孩子,并且这种认知思维是

很难自身发现的，最终发展成心理表征，这是国内家长教育的弊端。随着移动互联网的发展，需要一个平台提供给家长进行有效的沟通，进行咨询互动，形成以孩子为中心的交流社区。本应用则是应运而生，将教育与互联网技术的发展结合起来。

### 1.3 本文的论文结构和章节安排

本论文共分为六个章，每章节论述方向如下：

第 1 章 引言

主要描述研究背景和意义

第 2 章 相关理论和技术

介绍相关理论和最新计算机的技术以及移动通讯技术

第 3 章 系统分析

进行可行性分析和需求分析

第 4 章 系统的总体设计

介绍系统的架构设计，功能架构图，系统各模块的流程图

第 5 章 系统的详细设计

介绍各模块以及数据库的设计

第 6 章 系统的实现

通过图片以及文字的描述的形式展示系统的实现

第 7 章 系统测试

对系统进行各方面的功能测试，测试是否符合设计需求

第 8 章 总结与展望

总结本论文的心得感受

## 第 2 章 关键技术与工具

### 2.1 关键技术

#### 2.1.1 Vue.js 框架

Vue.js 是近两年来较为热门前端框架,用于构建应用页面展示,由国人龙雨溪开发,并持续维护更新代码库。在 Vue.js 出现之前,前端两大框架为 angular.js 与 react.js,有研究表明,Vue.js 无论从性能优化,页面渲染速度,开发环境的生态圈支持等,都胜于 angular.js 和 react.js,因此 Vue.js 具备 Angular.js 和 React.js 的优点,更重要的是,剔除了它们当中的缺点,而且官方有通俗易懂的教程,学习成本低,非常适合新手去尝试。Vue.js 本质是一个 JavascriptMVVM 库,通过内部封装形成一系列的 Api,只关注视图层,实现页面元素的响应式变化,其核心是数据驱动以及组件化开发。其次,Vue.js 支持丰富的第三方库的植入,促使本课题的开发进度更加高效,提供了更多的实现的可能性。

#### 2.1.2 Vuex 状态管理

Vuex 是一种基于 Vue.js 应用程序的状态管理模式技术。Vuex 能够抽离出系统组件中的共享状态对象,对状态对象,简单来说就是数据对象,进行集中管理,提供全局组件调用。任何共享的数据都可以以状态对象的形式存在,Vuex 的核心为 state(状态对象),view,action 主要三部分。state-view-action 形成单向数据流,state 驱动数据源渲染到 view 页面,view 页面通过 action 指令改变 state 状态,进而改变视图。随着如今应用开发日趋复杂,Vuex 的出现很好地解决了应用中冗余的请求数以及不同层级组件的复杂数据共享问题,使整体的状态变化更加清晰。

### 2.1.3 Koa

Koa 是基于 Node.js 的一个 web 框架，用于搭建应用的后台服务。随着 JavaScript 的快速发展，JavaScript 已经完全可以作为后端语言操作数据库为应用提供后台服务。Koa 作为中间服务控制器，受到众多开发者的青睐。在开发中可以应用 Koa 中间件服务，定义好业务层的数据逻辑，以对应的命名方式为前端页面提高完整的 API，使前端与后台的数据交流更加便捷和高效。在技术更新方面，Koa2.0 版本提供 async, await 语法支持，抛弃了传统 Node.js 异步镶嵌回调的痛苦，使异步的流程以一种同步的形式进行，在编写代码的时候更加便捷优雅。封装了 request 和 response 两个请求对象到全局上下文 context, 对开发者来说，方便获取到请求的参数以及设置请求成功返回的数据结构，非常适合用于本应用本地服务端的构建。

### 2.1.4 组件化开发

组件化开发是如今主流的一种应用开发工作模式。所谓组件化，就是提取有复用价值的东西，通过封装后，形成组件，暴露复用的 UI 和逻辑单元，对组件的粒度进行细分，可以理解为如下两种。一是 UI component (纯 UI 组件)，通过接受 props 作为数据的展示渲染，具有高度的复用性；而是 Logic component，这个可以理解为带有逻辑的 UI 组件，与数据进行交流。组件接受外部数据传输，组件化开发模式的出现，不仅很好地降低了复杂应用中的耦合度，更能大大提高应用的聚合度，使项目结构更加清晰，对项目的二次开发，后期维护以及项目优化起到了重要的作用。

## 2.2 相关工具

### 2.2.1 WebStorm 编辑器

WebStorm 是 JetBrains 的一个专门为 Web 开发人员设计的 IDE 代码编辑工



具。WebStorm 之所以能够稳站开发 IDE 之首，是以为它能够提供最业界最新技术的支持，智能化的标注提示能够让开发者在开发中提升了显著的工作效率。

### 2.2.2 Git 版本控制

Git 是世界上最先进的分布式版本管理系统。真正理解 Git 的意义，必须说起分布式。在分布式模式下，每个人的电脑都是一个完整的版本代码库，每个成员拥有属于自己的本地开发环境，进行各自所负责的功能模块的开发，最终把自己所完成的代码更新推送给对方，真正实现彼此分离又相对结合，因此可以轻松解决在多人协作方面出现代码不同步的问题。

### 2.2.3 Mock.js

随着技术的进步，当今的开发流程已有所改变，传统的开发流程是前端等待后端的接口，或通过自行模拟数据再进行开发。这种方式不仅对系统的功能需求或者工作效率都产生不好的影响，容易因为模拟的数据接口不对，导致功能的不完整和效率低下。Mock.js 的出现正是解决当中的弊端，可以自主提供完整的接口支持，模拟后端服务器接口的返回大量数据，相关字段，使前端能够在中摆脱后端因接口短缺的影响。在本应用中，将引进 Mock.js，返回相关数据进行页面开发以及调试，大大促进了开发的进程。

## 第 3 章 系统分析

### 3.1 研究目的

- (1) 学习对单页应用 APP 设计与开发的全过程
- (2) 对基于 Web 平台家长帮社区 App 进行全面深度开发。
- (3) 对实习期间所学的知识进行总结巩固

此次 web 平台家长帮互动社区 App 通过 Javascript 语言编写源代码, 在 Vue-cli 脚手架工具下搭建开发环境, 使用 webStrom IDE 开发出能应用到实际生活, 针对家长群体的一个互动交流应用。

### 3.2 可行性分析

可行性分析在开发任何一个软件应用项目的设计与实现的整个建设过程中占有十分重要的地位, 此外还会受到时间和周边资源的影响。所以在此接收任何一个开发项目时必须先进行可行性分析<sup>[11]</sup>。可行性分析主要对项目进行技术论证, 经济效益量度。对本应用进行可行性分析, 得出基于 Web 平台的家长帮互动 APP 设计与实现是否可能最终实现。可行性分析的目的是减少开发项目过程中所受到的风险, 避免在软件推进阶段出现人力物力的浪费, 使技术研发人员能够评估开发周期, 完成项目所需的人力, 从中能够获得的经济效益以及保证现在的技术能否支持和实现项目应用各功能的设计和开发。

#### (1) 技术可行性

技术可行性是指分析应用开发中所遇到的所有技术问题, 尽可能采取成熟稳定的技术提供软件开发。本课题应用前端基于 Vue.js 技术开发, Vue.js 已经作为目前单页 Spa 应用的技术主流, 拥有成熟的第三方组件支持, 可以搭建出优雅展示页面。后台基于 Node.js, 利用第三方框架 Koa+MySQL 的组合为应用提供后台服务。同时, 借鉴同类开源应用的实现写法, 结合本应用的实际功能, 可以觉得本应用所遇到技术层面所遇到的问题。

### （2）经济可行性

本应用采用 web 平台开发，对开发的硬件设备要求不高，而且应用在设计以及研发上面都是由本人负责，无需支付额外的设计和开发费用，因此实现本应用的功能的经济成本是可控的，没有过多的经济资源的浪费。

### （3）系统可行性

本课题设计的应用基于 web 平台，可以在所有 PC 电脑上面运行，同时，本应用可以通过第三方 IDE 最终打包成 Android 或 iOS 手机应用，以当前只能手机的配置性能，完全可以支持本应用的安装和运行。

综上所述，“基于 Web 平台的家长帮 APP 的设计与实现”参考上面三点的可行性详细分析，得出本应用是完全可行的结论。

## 3.3 需求分析

需求分析是建立软件期间对软件的可行性分析进行求精和细化。对最终实现的软件应用提出完整，准确，清晰，具体的要求，是软件工程中重要的一部分。如果需求分析不准确，对后期的开发的影响是不可以预估的，因此，需求分析不可急于求成，需要精确严谨。需求分析要求详细描述系统应用的功能以及具体模块的详细描述。

### 3.3.1 功能模块需求分析

家长帮社区应用以家长互动交流为核心，为家长提供一个互动交流的社区平台，力求保证家长能在本平台获取真实有效的咨询，为家长在这个社区平台互动能够在育儿，养儿，教儿过程中能够获取相关咨询，帮助孩子更好地发展；同时本应用加入家长头条模块，满足家长的基本日常需求，提高家庭生活质量。

本应用建立与孩子为话题的社区交流互动社区。主要功能有浏览社区用户帖子，评论互动，发表话题帖子，记录孩子印记，菜谱教程，用户信息管理等，具体模块实现如下：

（1）社区：对帖子进行评论互动，帖子类型有问答，分享，日常等三大类型作。

(2) 成长广场：以孩子的成长历程为话题，建立中心式的家长分享群体，家长可以在此浏览其他用户的成长记录，进行点赞，评论。

(3) 家长头条：获取头条信息，在本模块提供生活小知识，小技能，丰富家长的家庭生活。

(4) 个人信息：编辑自己的个人信息，查看发布，收藏，点赞的贴子，发出的评论，同时，提供孩子的档案记录。

### 3.3.2 系统功能

#### (1) 用户注册

游客通过校验手机、填写正确的验证码可以注册成为用户。用户可以在社区进行互动交流。

#### (2) 用户登录

游客输入账号、密码，核对正确之后即可登陆应用，进行互动交流和修改个人信息。

#### (3) 社区贴子浏览

游客和用户可以在头条和社区论坛模块浏览详细内容，社区论坛将展示贴子的标题，类型，作者头像和姓名，发贴时间以及浏览次数。

#### (5) 评论留言

用户登陆后可以对社区论坛的贴子评论，进行互动交流，该功能对游客不开放。

#### (6) 发布话题贴子

用户可以在发贴模块选择贴子主题，编辑好标题，内容后进行发布，发布成功后的贴子将会展示在首页浏览区。

#### (7) 成长广场

用户在本模块可以记录孩子的成长时光，同时也可以浏览查看别的用户的成长记录。

#### (8) 个人中心

用户在此模块可以修改相关基本信息，如昵称，性别，头像等。用户在此模

块修改孩子档案，查看成长孩子成长历程等。

### 3.3.3 关键用例图及用例规约

#### (1) 用户注册用例规约

表 3-1 用户注册用例规约表

用例编号	ZWFW001
用例名称	用户注册
参与者	游客
用例描述	该用例的功能是让游客注册成为平台的一个用户
前置条件	用户须要正确校验手机验证码
后置条件	用户填写相关的注册信息以及正确的手机验证码后，点击注册，系统将用户信息录入到数据库中
(续上表)	
基本流程	1) 用户填写手机，点击下一步获取验证码 2) 填写验证码后点击下一步，系统后台检测验证码是否正确，正确输入才能进入基本注册信息页面 3) 填完基本注册信息，校验通过后提交完成注册
备选流程	用户在注册过程中如果信息不符合规则，则无法通过校验，需重新填写

#### (2) 用户登录用例规约

表 3-2 用户登录用例规约表

用例编号	ZWFW002
用例名称	用户登陆
参与者	用户
用例描述	该用例的功能是用户进行登陆操作
前置条件	在点击进入发贴模块，信息中心以及个人中心时，会跳转至登录页面
后置条件	用户填写账号和密码后，点击登录，正确则登录成功
基本流程	1) 用户选择进入登陆界面，输入账号和密码，点击按钮进行登录 2) 系统后台根据用户填写的信息进行检测，如果账号存在则将对向数据库进行查询校验 3) 输入账号和密码一致，则登录系统，不一致则提示错误信息并要求用户重新输入

(续上表)

备选流程	用户输入的账号不存在或者密码错误时,应用提示用户登录失败并要求重新输入
------	-------------------------------------

### (3) 用户评论

表 3-3 用户评论用例规约表

用例编号	ZWFW003
用例名称	用户评论
参与者	用户
用例描述	该用例的功能是用户对贴子进行评论
前置条件	用户已通过账号密码成功登陆应用
后置条件	用户完成评论输入, 点击评论按钮
基本流程	1) 用户点击进入社区论坛模块, 点击进入贴子详情, 点击评论按钮 2) 出现评论输入框, 完成评论后点击确定 3) 系统后台根据用户填写的评论进行检测, 检测内容格式是否正确。如果不正确, 提示用户重新输入, 评论成功后将会在贴子下面出现本评论, 并显示评论者相关名字。
备选流程	如果用户在点击评论按钮前不是登陆状态, 则跳转到应用登陆界面。

### (4) 用户发贴

表 3-3 用户发贴用例规约表

用例编号	ZWFW004
用例名称	用户发贴
参与者	用户
用例描述	该用例的功能是用户发布贴子内容
前置条件	用户已通过账号密码成功登陆应用
后置条件	用户完成标题, 类型, 内容的填写, 点击发布按钮

(续上表)

基本流程	1) 用户进入社区论坛模块 2) 点击下方发贴按钮, 输入标题, 选择类型, 编写好贴子主体内容 3) 用户确认发布贴子 4) 系统后台根据用户填写的标题, 内容进行检测, 是否存在敏感字段。 5) 如果检测失败, 提示用户重新输入 6) 检测通过, 将该贴子的相关信息保存在数据库, 并展示到主页浏览区。
备选流程	如果用户在点击发布前按钮前不是登陆状态, 则跳转到应用登陆界面。

## (5) 查看互动消息

表 3-5 查看互动信息用例规约表

用例编号	ZWFW005
用例名称	查看互动信息
参与者	用户
用例描述	该用例的功能是用户进入个人信息模块并查看互动信息
前置条件	用户已通过账号密码成功登陆应用
后置条件	用户点击返回
基本流程	1) 用户点击进入个人信息模块 2) 用户点击进入消息中心 3) 在消息中心中可查看互动消息 4) 用户可对互动消息进行回复 5) 回复流程参照用户评论流程 6) 回复成功后将更新互动信息列表
备选流程	如何无互动消息, 页面将展示为空

## (6) 修改个人信息

表 3-6 修改个人信息用例规约表

用例编号	ZWFW006
用例名称	修改个人信息
参与者	用户

(续上表)

用例描述	该用例的功能是用户修改个人信息
前置条件	用户已通过账号密码成功登陆应用
后置条件	用户点击完成按钮
基本流程	1) 用户进入个人信息模块，点击修改按钮 2) 出现修改个人信息页面，输入可修改昵称，性别，头像，孩子出生日期。 3) 应用将对用户输入的信息进行检测，如有格式不符合要求，则提示用户重新输入 4) 点击保存按钮，检测修改内容通过后，将在数据库更新用户的个人信息。
备选流程	用户在此过程中如果信息不符合规则，则无法修改，需重新填写

(7) 添加孩子成长印记

表 3-7 添加孩子成长印记用例规约表

用例编号	ZWFW007
用例名称	添加孩子成长印记
参与者	用户
用例描述	该用例的功能是用户添加孩子成长印记
前置条件	用户已通过账号密码成功登陆应用
后置条件	用户点击完成按钮
基本流程	1) 用户进入成长广场模块， 2) 点击记录按钮， 3) 页面展示孩子成长记录内容，点击添加， 4) 填写成长印记文字内容， 5) 点击保存按钮，更新页面成长记录历程。
备选流程	用户输入的成长印记主题过长，则提示用户重新输入

(8) 查看孩子成长历程

表 3-8 查看孩子成长历程用例规约表

用例编号	ZWFW008
用例名称	查看孩子成长历程
参与者	用户

(续上表)



(续上表)

用例描述	该用例的功能是用户查看孩子成长历程
前置条件	用户已通过账号密码成功登陆应用
后置条件	用户点击返回按钮
基本流程	1) 用户进入个人信息模块, 2) 用户点击孩子成长历程模块, 3) 页面展示孩子成长历程 4) 用户选择年份进行显示孩子成长历程
备选流程	如果用户成长记录为空, 则页面提示用户为空, 是否要开始进行记录

### 3.4 系统开发环境

#### (1) 软件需求

开发工具: WebStorm、WPS Office、Git

开发语言: Javascript

需求分析工具: Rational Rose

#### (2) 硬件需求

表 3-3 硬件需求表

类别	电脑	手机
名称	联想笔记本	苹果
型号	Intel 7	iPhone 6s plus
操作系统	Windows 10	iOS 10.3.2

# 第 4 章 系统的总体设计

## 4.1 系统架构设计

本应用的架构设计如下图所示，详细分析了应用的展示层，服务层，应用层的设计：

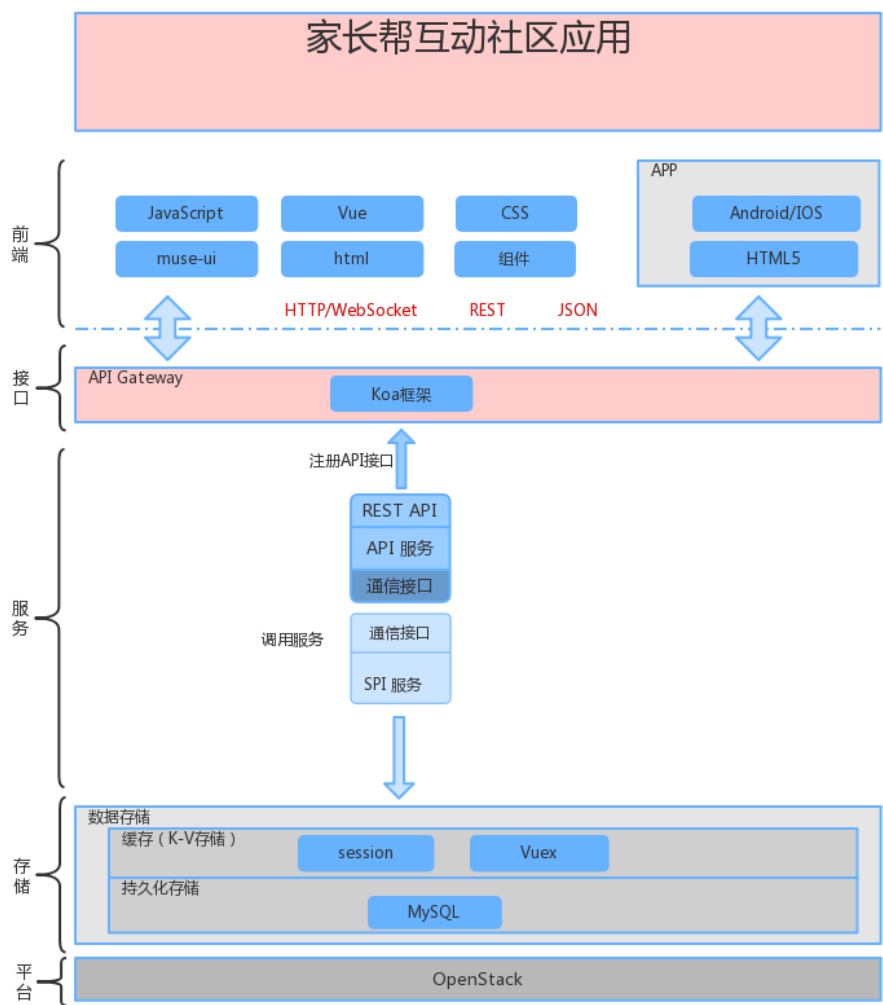


图 4-1 系统架构图

本应用的架构设计将从前端，接口，服务，存储，平台等方面进行分析。前端页面展示使用 Html，Css，JavaScript 等基本语言来构建应用的展示页面，为了让应用的设计更加和谐美观了，特意引入 Muse-UI 第三方库进行组件化开发，

大大提高开发的效率和应用的质量；后台服务通过 Node.js 的 Koa 框架连接数据库进行接口的调配，根据应用的功能设计出相应的接口 Api 给前端页面调用。由于本应用存在了多种数据状态，为了让数据状态更加容易管理，特意引入了 Vuex 的数据管理机制，将数据状态全局化，使组件各个组件能够共享数据。同时，对用户的登录状态用 session 缓存技术记录下来，保证了一定的时效性。对应用的内容数据以及用户信息，将用 mysql 数据库存储在本地硬盘，保证数据不轻易丢失。

## 4.2 系统功能架构图

本应用的设计采用图形方法解析它的功能与流程，下图是本应用的系统功能架构图：

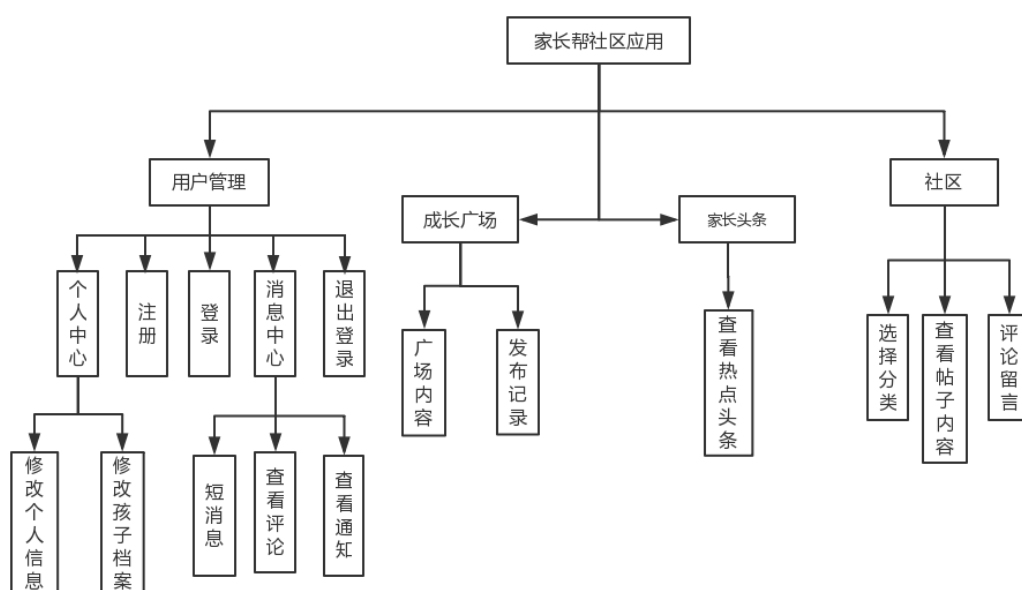


图 4-2 家长帮互动社区 APP 整体功能结构图

## 4.3 系统各模块流程图

### 4.3.1 发布贴子流程图

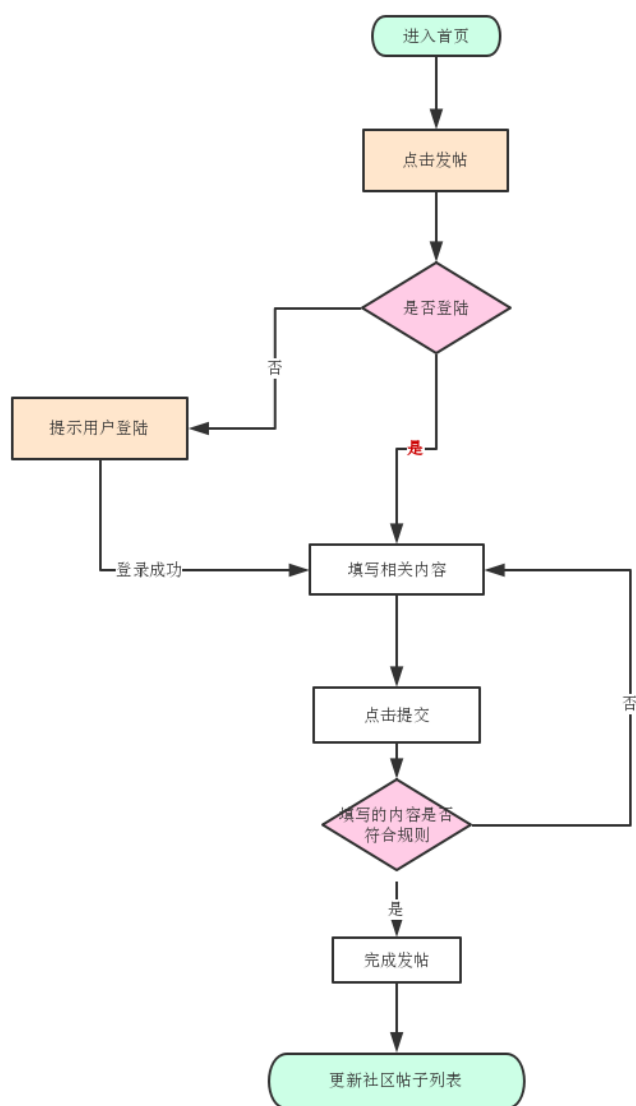


图 4-3 发布贴子流程图

用户点击发帖，系统检测用户是否已经处于登录状态，如果没有，弹出登录/注册模态窗提示用户，在发贴操作前应该先进行登录，未注册的游客也可以在此通过手机号码进行注册成为用户。用户通过账号密码登录成功后进入贴子内容编辑页面，用户需要编辑标题，详细内容以及贴子类型，填写完成后，点击发布，应用系统监测填写内容是否有敏感词语或者不合法格式，如果内容符合要求，贴子发布成功，返回首页，并更新首页内容；否则，则提示用户需要重新编辑。

### 4.3.2 社区模块流程图

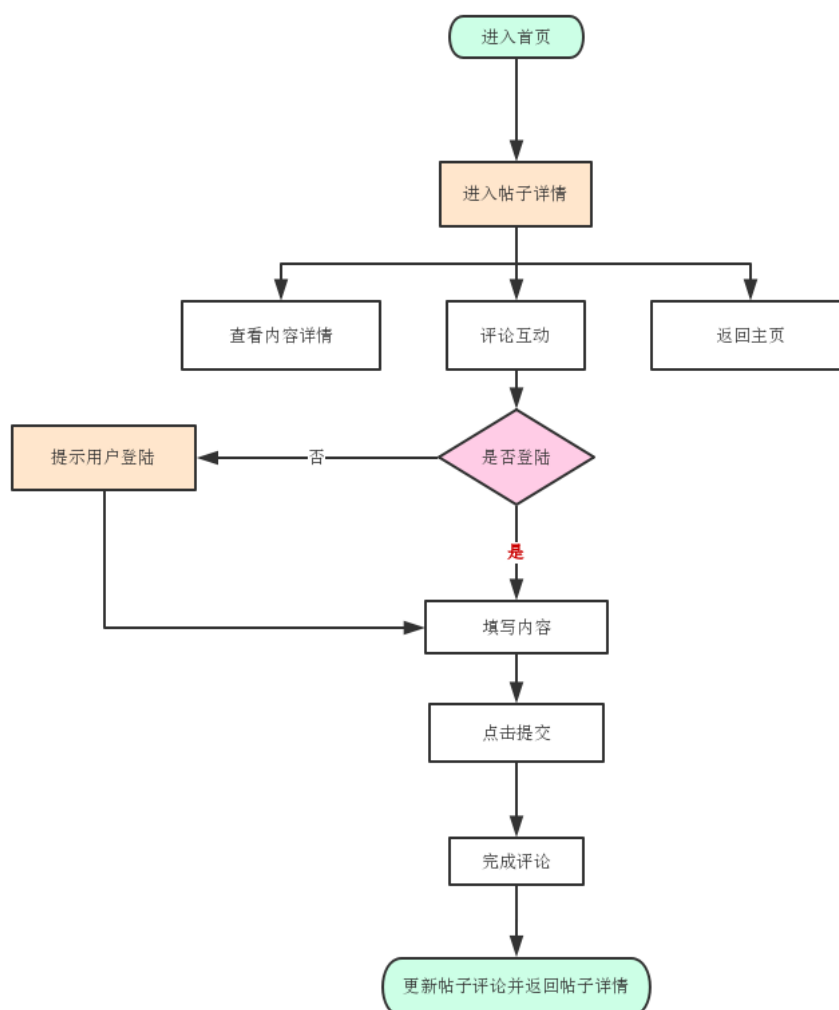


图 4-4 社区模块流程图

用户打开应用，即可进入社区模块，社区模块为应用的首页，用户可以对帖子列表进行浏览，帖子在首页中会展现贴子的作者，帖子类型，浏览次数，发布时间等基本信息。用户在浏览过程中，可以点击某一帖子展示具体内容，用户进入帖子详细内容后，可以查看详细内容以及相关评论信息。用户可以对帖子作者发起评论或者对某一评论进行评论。用户评论互动时，需要填写相关评论内容，点击提交评论，如系统检测内容符合要求并且没有敏感词语，则评论成功，通知相关评论的参与者，更新评论内容。用户如不进行互动，可以点击返回首页。

### 4.3.3 成长广场流程图

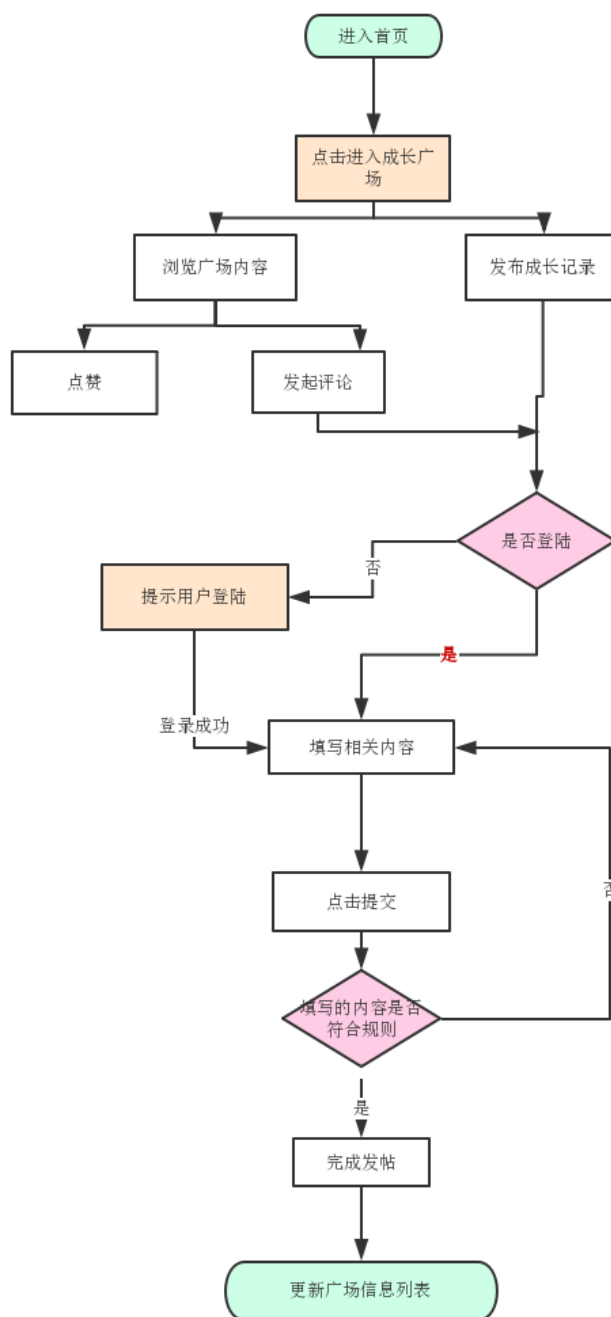


图 4-5 成长广场流程图

用户可以在应用底部导航栏点击进入成长记录模块。成长记录模块目的为家长打造一个孩子成长印记交流群体社区。用户可以在该模块可以查看其他家长的孩子记录，用户必须以成长为主题发布印记，以文字+图片为载体进行记录。用

户可以对某一印记进行点赞和评论，评论流程与社区评论流程一致。用户可以对自己的孩子进行成长印记，发布流程与发布贴子一致。

#### 4.3.4 个人模块流程图

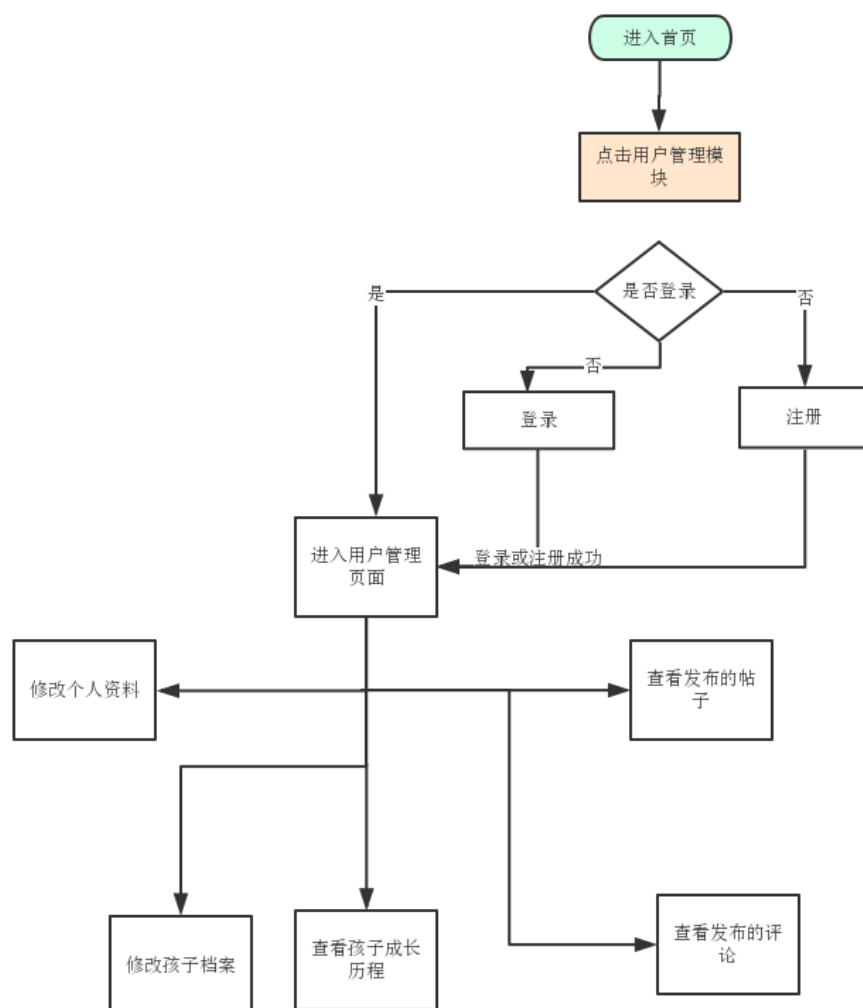


图 4-6 个人模块流程图

用户点击进入个人模块，个人模块是对用户的信息进行集中处理。用户在点击个人模块时，应用系统检测用户是否已经处于登录状态，如果还没登录，则弹出登录/注册模态窗口提示用户需要进行登录注册。登录成功后，进入个人模块首页，展示用户的基本个人信息，提供修改个人信息入口。用户可以在此查看已发布的贴子，已发布的评论，以及已收藏的贴子；个人信息模块提供孩子的档案

记录，用户可以在此编辑相关孩子出生信息，如出生年月与孩子的关系，孩子的昵称等，该档案的信息会暴露到贴子详情和成长广场当中。用户可以查看孩子的成长印记，以时间戳排序展示。



## 第 5 章 系统详细设计

### 5.1 系统模块介绍

通过对家长帮社区应用进行需求描述、领域知识、数据建模以及成熟应用的理解和借鉴，得到本系统的关键抽象，即析取出系统的参与活动的各个类的具体化。

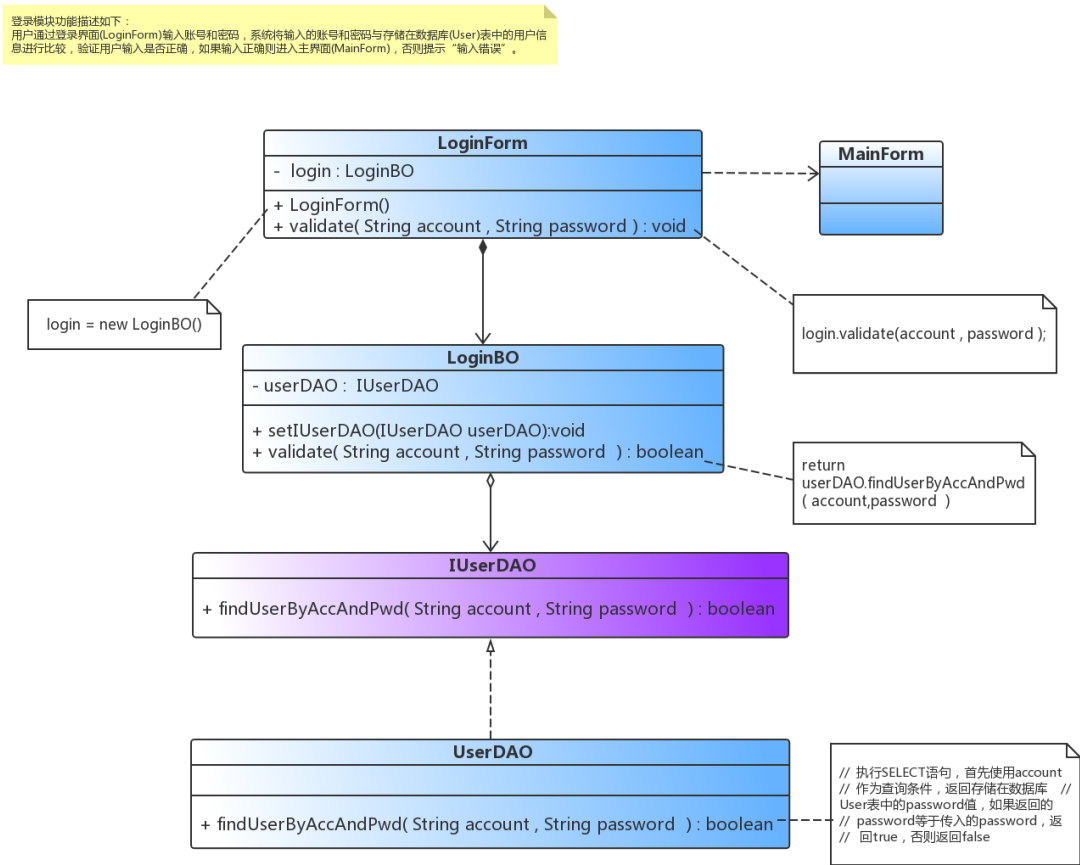
#### 5.1.1 系统实体类

本系统设计有四个实体对象，分别是：

- 1) 用户实体，包含成功注册的用户对象的信息。
- 2) 社区贴子实体，包含用户发布贴子的信息，每个用户对应多个贴子实体对象。
- 3) 成长记录实体，包含用户对孩子成长历程，每个用户只对应多个成长记录实体。
- 4) 头条实体，包含头条详细信息。

#### 5.1.2 系统部分控制类

由于系统应用控制类较多，因此挑选用户登录模块控制类进行详细分析，具体如下所示：



类说明：

类 名	说 明
LoginForm	登录窗口，省略界面组件和按钮事件处理方法（边界类）
LoginBO	登录业务逻辑类，封装实现登录功能的业务逻辑（控制类）
IUserDAO	抽象数据访问类接口，声明对用户表的数据操作方法，省略除查询外的其他方法（实体类）
UserDAO	具体数据访问类，实现对User表的数据操作方法，省略除查询外的其他方法（实体类）
MainForm	主窗口（边界类）

方法说明

类 名	说 明
LoginForm类的LoginForm()方法	LoginForm构造函数，初始化实例成员
LoginForm类的validate()方法	界面类的验证方法，通过调用业务逻辑类LoginBO的validate()方法实现对用户输入信息的验证
LoginBO类的validate()方法	业务逻辑类的验证方法，通过调用数据访问类的findUserByAccAndPwd()方法验证用户输入信息的合法性
LoginBO类的setIUserDAO()方法	Setter方法，在业务逻辑对象中注入数据访问对象（注意：此处针对抽象数据访问类编程）
IUserDAO接口的findUserByAccAndPwd()方法	业务方法声明，通过用户账号和密码在数据库中查询用户信息，判断该用户身份的合法性
UserDAO类的findUserByAccAndPwd()方法	业务方法实现，实现在IUserDAO接口中声明的数据访问方法

图 5-1 登录模块类图

### 5.1.3 系统时序图

#### 1) 查看贴子

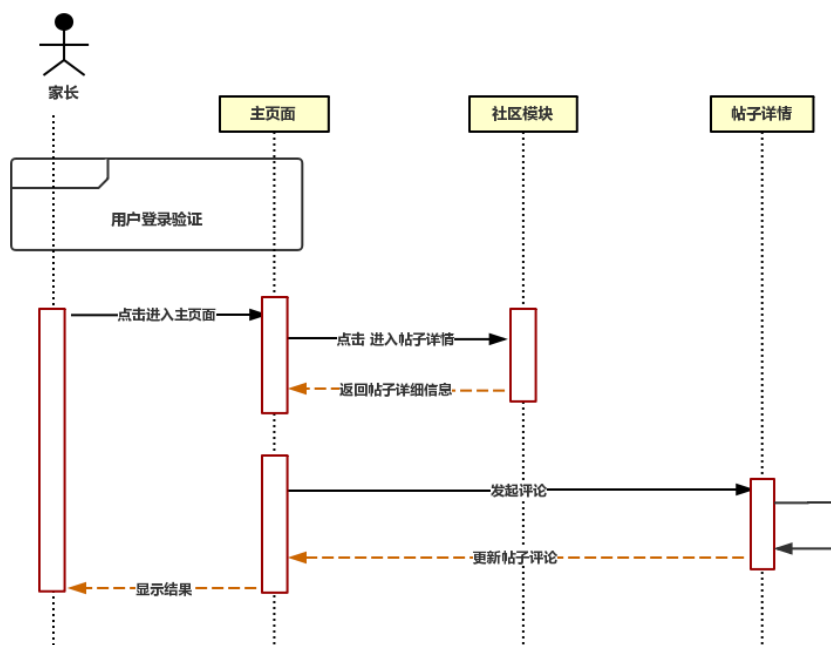


图 5-2 查看贴子时序图

#### 2) 用户登录

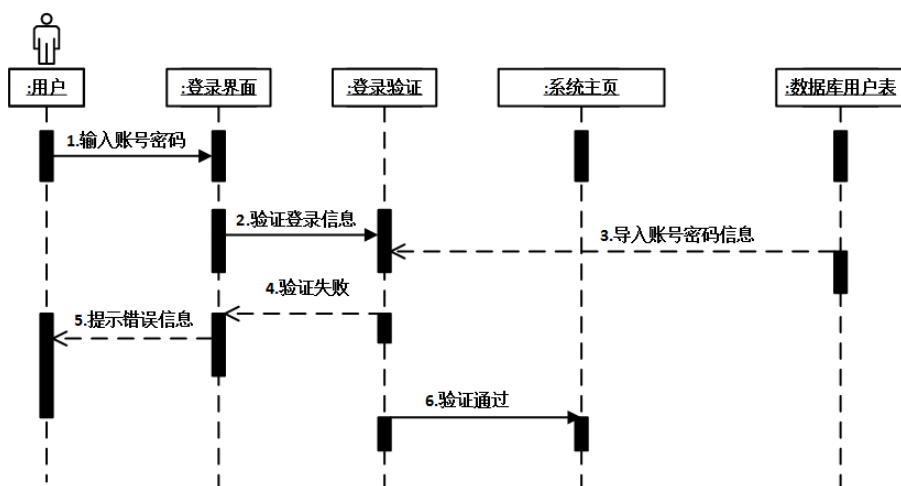


图 5-5 用户登录时序图

#### 3) 用户注册

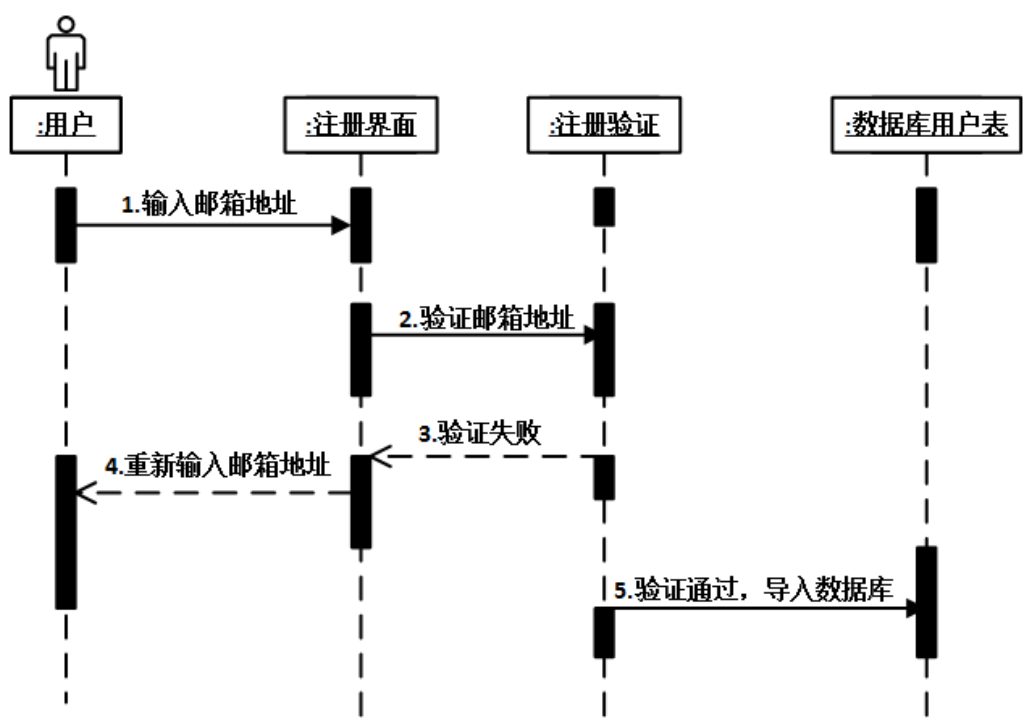


图 5-6 用户注册时序图

5.2 系统数据库设计

数据库详细设计展示数据库表的详细设计以及各表数据的字段约束，具体如下图所示：

(1) 用户个人信息表

用户个人信息表用于用户的基本信息管理，因此该表约束了每个字段的字长，字符类型。需要注意一点，在表中 uid 字段设计为用户的唯一标识，系统可根据该字段标识定位到单一用户，并获取到相关具体数据，具体如下图所示：

Objects   user @parent_help (liao) - T...   Untitled @parent_help (liao)...   * user @parent_help (liao) ~...									
Save   Add Field   Insert Field   Delete Field   Primary Key   Move Up   Move Down									
Fields	Indexes	Foreign Keys	Triggers	Options	Comment	SQL Preview			
Name	Type	Length	Decimals	Not null	Virtual	Key	Comment		
uid	int	11	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	1			
user_name	varchar	50	0	<input type="checkbox"/>	<input type="checkbox"/>				
password	varchar	50	0	<input type="checkbox"/>	<input type="checkbox"/>				
gender	tinyint	2	0	<input type="checkbox"/>	<input type="checkbox"/>				
phone	varchar	12	0	<input type="checkbox"/>	<input type="checkbox"/>				
address	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>				
avatar	varchar	100	0	<input type="checkbox"/>	<input type="checkbox"/>				

图 5-7 个人信息表

贴子详情表

贴子详情表用于贴子的信息管理，因此该表约束了每个字段的字长，字符类型。需要注意一点，在表中 pageId 字段设计为用户的唯一标识，系统可根据该字段标识定位到具体某一贴子，并获取到相关具体数据，同时通过 authorId 关联作者信息，具体如下图所示：


Fields	Indexes	Foreign Keys	Triggers	Options	Comment	SQL Preview					
Name				Type	Length	Decimals	Not null	Virtual	Key	Comment	
▶ pageId				int	11	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1		
authorId				int	11	0	<input type="checkbox"/>	<input type="checkbox"/>			
type				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			
title				varchar	50	0	<input type="checkbox"/>	<input type="checkbox"/>			
content				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			
visit_count				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			
comments				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			
pics				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			

图 5-8 贴子信息表

1) 成长记录表

成长记录表展示成长记录的具体数据字段，其中 growId 为该表唯一标识，不可重复，authorId 关联成长记录的作者，具体设计如下：

Fields	Indexes	Foreign Keys	Triggers	Options	Comment	SQL Preview					
Name				Type	Length	Decimals	Not null	Virtual	Key		
I growId				int	11	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
authorId				int	11	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>			
title				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			
content				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			
pics				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			
comment				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			
favorites				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>			

图 5-9 成长记录表

2) 头条信息表

头条信息表具体设计如下，其中 from 字段为头条信息来源的唯一标识。

Fields	Indexes	Foreign Keys	Triggers	Options	Comment	SQL Preview				
Name				Type	Length	Decimals	Not null	Virtual	Key	
newId				int	11	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
title				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
▶ content				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
from				varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		

图 5-10 头条信息表

## 第 6 章 系统的实现

### 6.1 开发环境搭建

本应用开发基于 Node.js 环境, Node.js 的 NPM 工具是目前业界最大的生态系统。Node.js 的特性是单线程, 非阻塞以及事件循环。在 Node.js 环境下可以通过 NPM 管理工具安装和维护第三方依赖库, 详细实现如下:

#### 6.1.1 安装 Node 环境

##### (1) 下载 Node.js 安装包

进入 <https://nodejs.org/en/download/> 下载页面, 选择系统对应的安装包, 基于本人电脑配置, 选择了 window x64 位版本, 下载页面如下图所示:

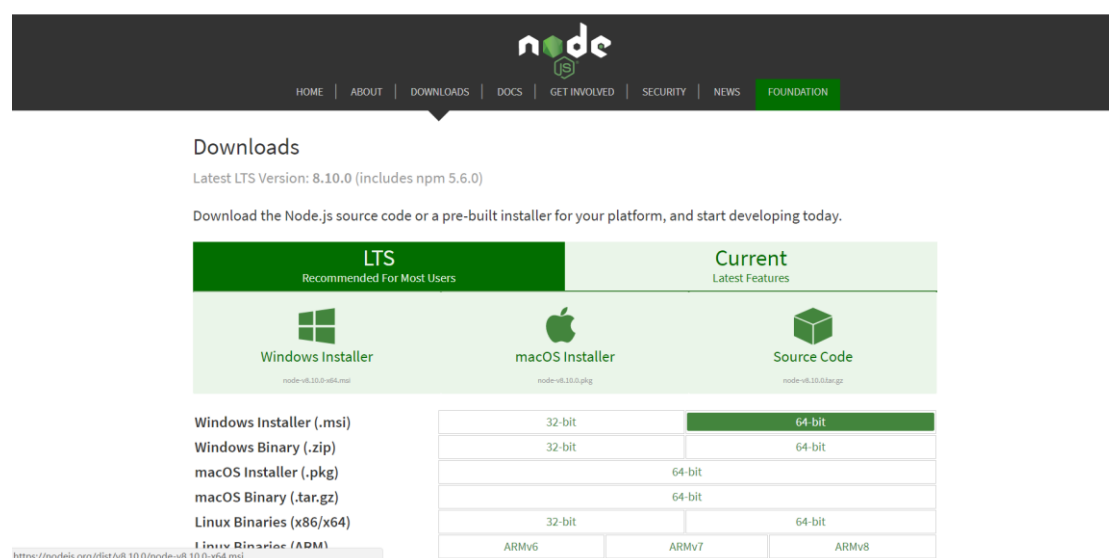


图 6-1 Node.js 下载页面图

##### (2) 安装

等待下载完成, 双击下载好的文件进入安装流程, 由于安装流程也不见智能化, 根据提示一直点 next 下一步即可完成安装, 于是不再详细描述安装流程。

安装完成后将带有 NPM 管理安装包，安装完成后如下图所示：

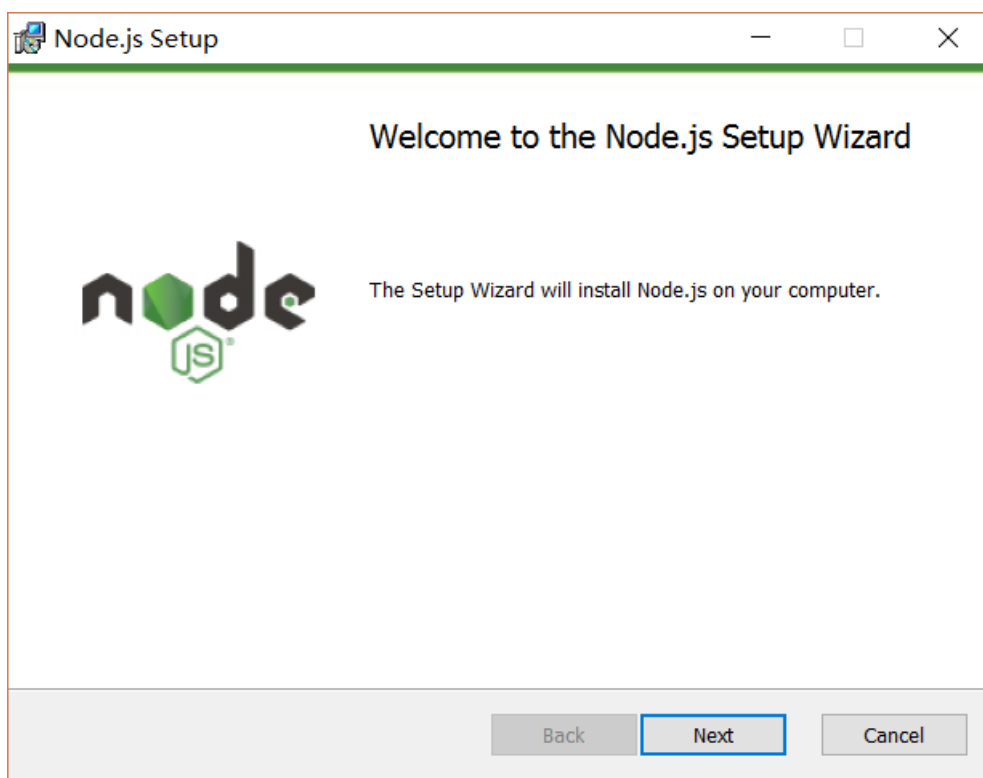


图 6-2 安装完成图

### (3) 检查是否安装成功

Window 下环境，通过 win + r 组合快捷键调出系统命名窗口，输入 cmd 进入控制台，分别输入命令“node -v”，“npm -v”，查看本机的 node 环境版本和 npm 版本，如果成功安装，将出现如下版本信息：

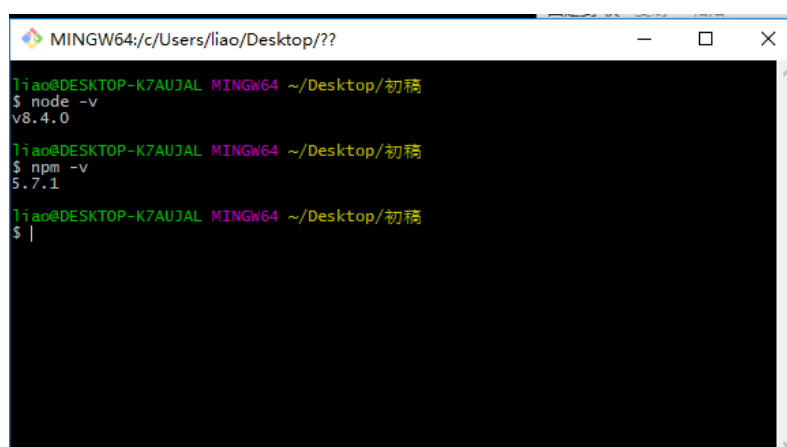


图 6-3 本机 node 环境版本信息

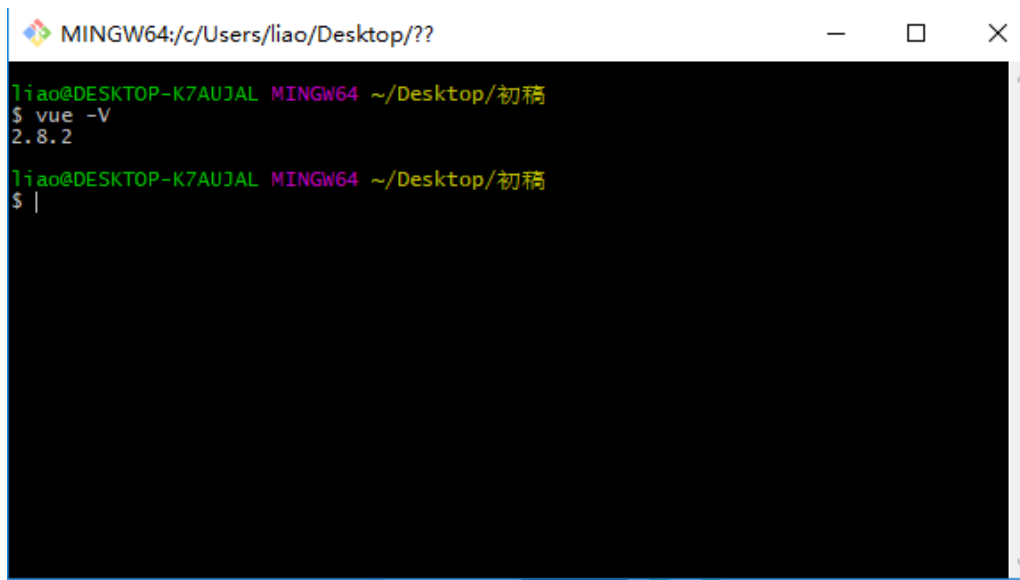


### 6.1.2 快速搭建项目

本应用将通过 vue-cli 搭建项目目录结构，vue-cli 的详细介绍已经在第二章论述，将从安装构建以及如何构建项目这两点论述，具体实现如下：

#### (1) 安装 Vue-cli

调出 cmd 系统命令窗口，通过命令“npm install -g vue-cli”进行安装，这个命名是 npm 安装依赖包的一个模板，vue-cli 为工具包的名称，-g 表示全局安装。安装成功后可以通过“vue -V”查看 vue-cli 的版本，只要命令中 V 必须为大写，否则命令行将会提示无法识别。下图为本应用的 vue-cli 版本：



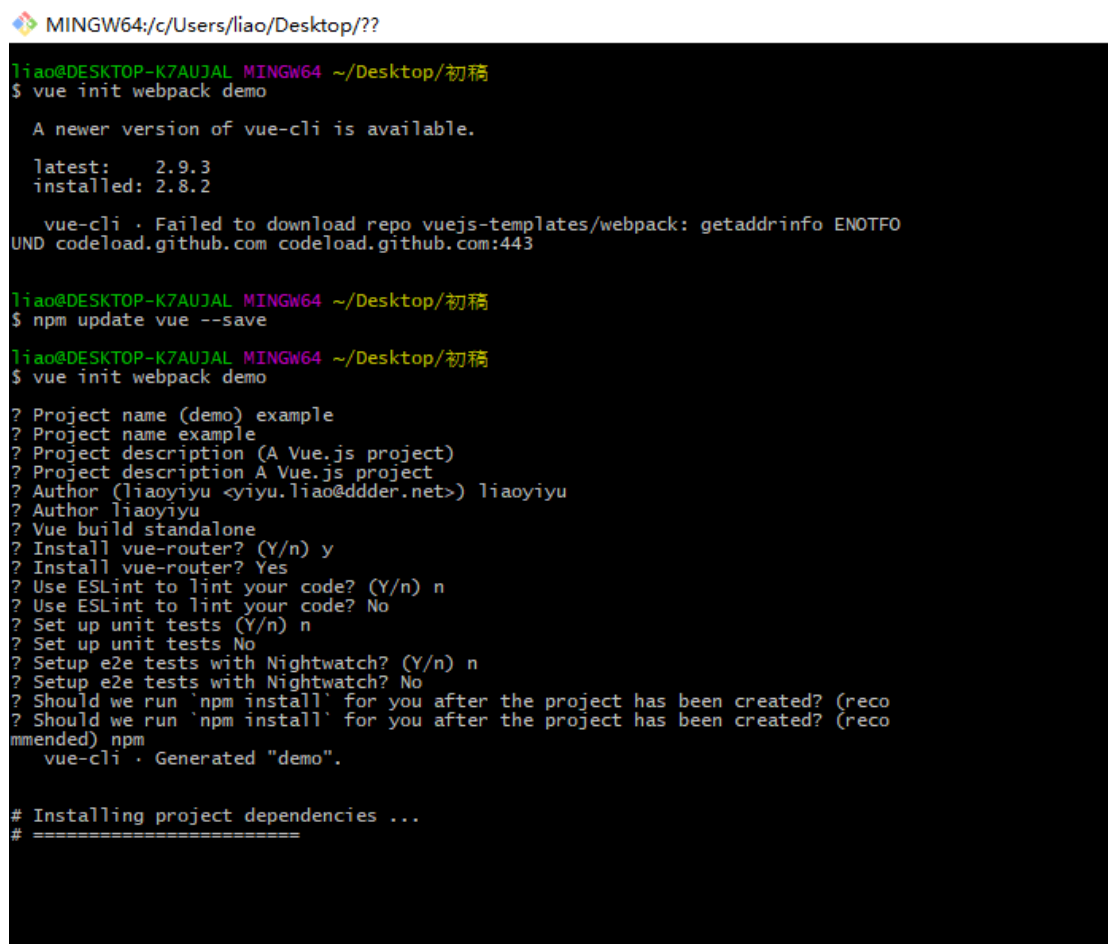
```
MINGW64: c:/Users/liao/Desktop/??
liao@DESKTOP-K7AUJAL MINGW64 ~/Desktop/初稿
$ vue -V
2.8.2
liao@DESKTOP-K7AUJAL MINGW64 ~/Desktop/初稿
$ |
```

图 6-4 本机 vue-cli 版本信息

#### (2) 构建项目

安装好 vue-cli 环境后，就可以快速地搭建项目结构了，对于简单项目来说，这个意义在于能够规范化项目结构，形成一个代码结构规范，同时，对于开发者来说，能够省下很多时间，让开发者专注与项目的研发当中。进入系统命令 cmd 窗口通过命令“vue init webpack project\_name”构建项目，如果当前系统的 vue-cli 版本不是最新，可以通过“npm update vue --save”进行版本的更新。命名输入正确，系统将以智能化的形式提示输入

应用的名字，简单的描述，作者，以及根据需要是否安装安装依赖包，下图为本应用的构建流程：



```
MINGW64:/c:/Users/liao/Desktop/??  
liao@DESKTOP-K7AUJAL MINGW64 ~/Desktop/初稿  
$ vue init webpack demo  
  
A newer version of vue-cli is available.  
  
latest: 2.9.3  
installed: 2.8.2  
  
vue-cli . Failed to download repo vuejs-templates/webpack: getaddrinfo ENOTFO  
UND codeload.github.com codeload.github.com:443  
  
liao@DESKTOP-K7AUJAL MINGW64 ~/Desktop/初稿  
$ npm update vue --save  
  
liao@DESKTOP-K7AUJAL MINGW64 ~/Desktop/初稿  
$ vue init webpack demo  
  
? Project name (demo) example  
? Project name example  
? Project description (A Vue.js project)  
? Project description A Vue.js project  
? Author (liaoyiyu <yiyu.liao@ddder.net>) liaoyiyu  
? Author liaoyiyu  
? Vue build standalone  
? Install vue-router? (Y/n) y  
? Install vue-router? Yes  
? Use ESLint to lint your code? (Y/n) n  
? Use ESLint to lint your code? No  
? Set up unit tests (Y/n) n  
? Set up unit tests No  
? Setup e2e tests with Nightwatch? (Y/n) n  
? Setup e2e tests with Nightwatch? No  
? Should we run 'npm install' for you after the project has been created? (reco  
mmended) npm  
vue-cli . Generated "demo".  
  
# Installing project dependencies ...  
# =====
```

图 6-5 项目构建图

## 6.2 安装关键依赖库

进入项目的目录下，通过“npm install”命令，把所有库环境安装注入到项目中。打开项目目录中的 package.json 文件，package.json 文件为项目的描述文件，在此可以看到一个项目的基本描述信息。在 dependencies 中可以看到应用所依赖的相关库。具体如下图所示：

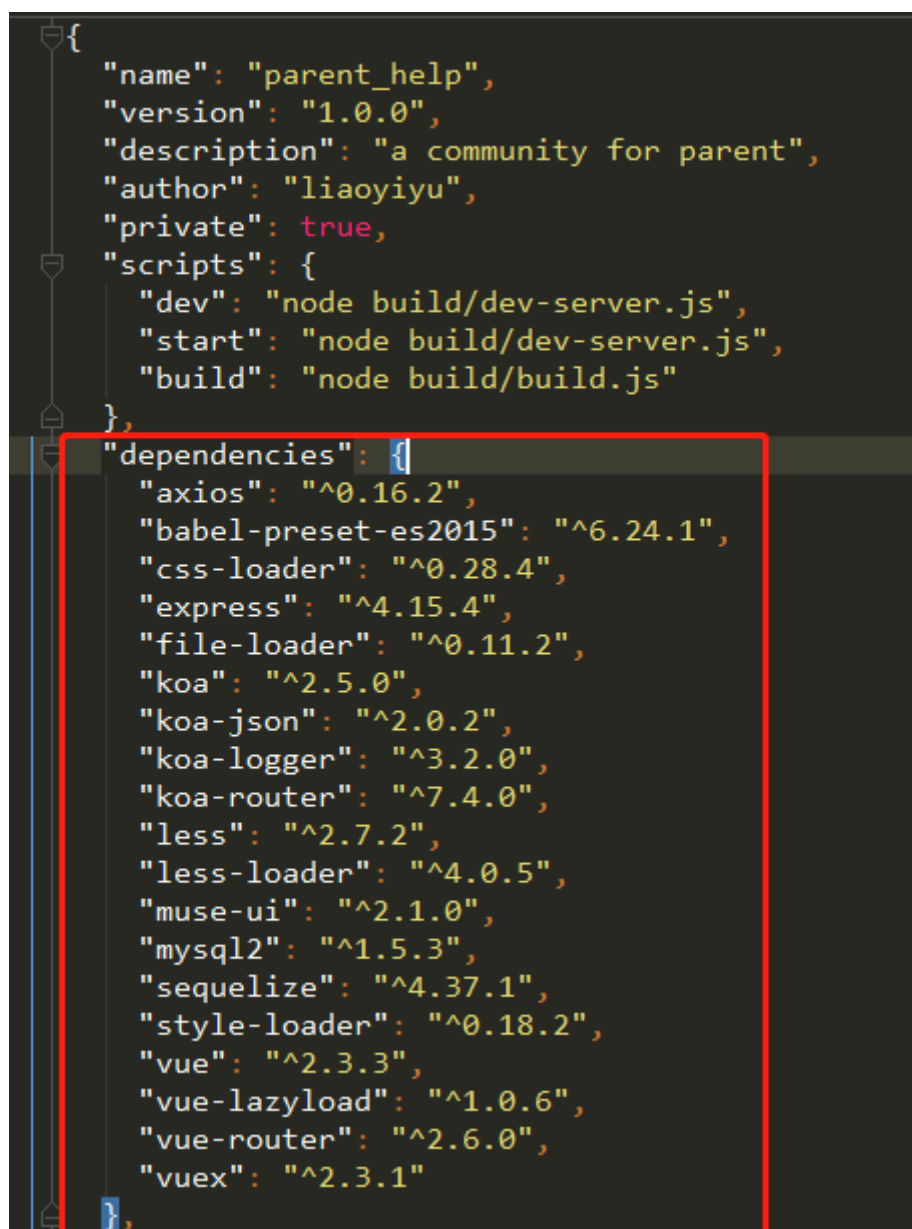


图 6-6 项目相关描述文件图

### 6.3 工程结构说明

在设计开发阶段，确定以 Parents\_Help 为工程名，构建好每个文件的位置以及命名规则，方便后期维护，仔细分析每一行代码需要实现的功能，一步步地构建起整个应用。

由于项目由 vue-cli 构建，基于 webpack 环境，因此将重点放在 src 文件夹

和 service 文件夹中。src 文件夹为应用的前端工程页面源代码，service 为基于 node 环境的后端服务源代码。具体结构如下图所示：

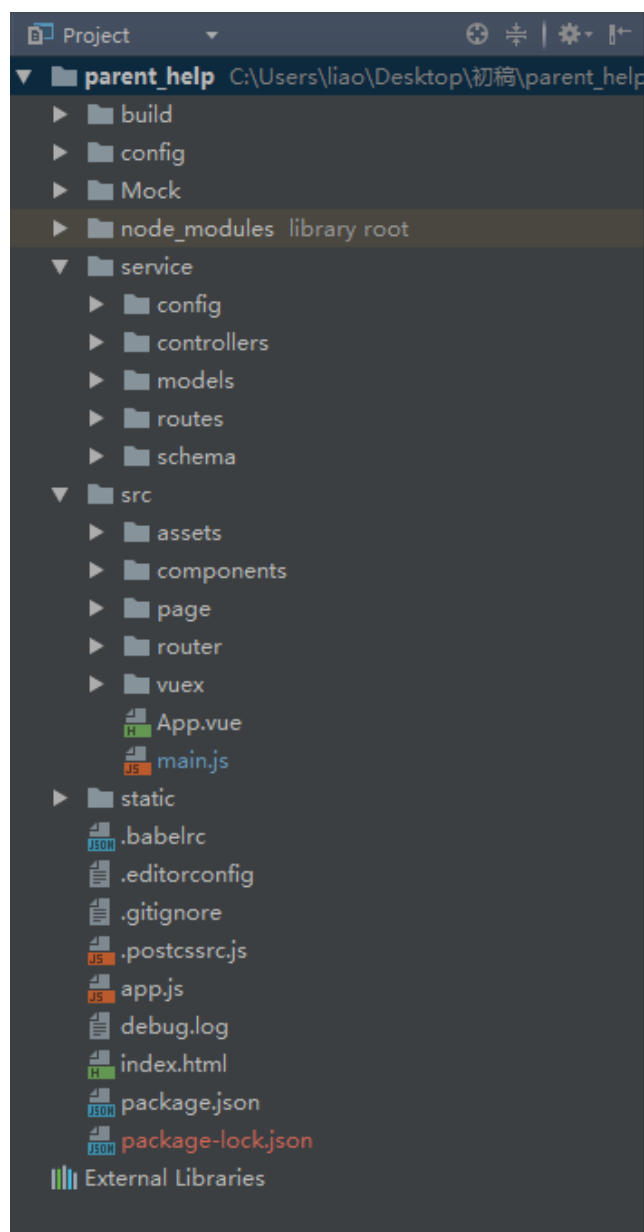


图 6-7 工程目录图

## 6.4 系统页面效果

介绍完系统的开发环境后，接下来将介绍各模块功能的实现，具体如下：

1) 社区界面

社区首页分为顶部，中部以及底部三个部分。顶部展示应用的名字应用的标识 logo，同时顶部右上角提供信息中心入口，用户点击该入口可进入消息中心，查看动态消息。中部为展示社区所有贴子列表区域，可以浏览每个贴子中作者的头像，地址，昵称，贴子发布时间，点赞量，评论数以及浏览次数等基本信息。用户可以在根据年级学段选择社区分类，更新社区列表内容。在切换过程中，在前端与后台获取数据的中间，会出现加载提示用户，提高用户的使用体验。底图为应用的导航栏，分别为首页，成长广场，发布，头条，用户个人信息管理等五个模块，点击不同模块应用将更新展示内容。具体实现如下图所示：



图 6-8 社区首页



图 6-9 加载提示



图 6-10 选择“小学”类别

## 2) 成长广场

## 3) 头条

头条模块为家长提供生活知识，可以理解为家长的成长教堂，目的是为了让家长获取更多优质的咨询，以文章推送的形式进行。在本模块分为三部分，顶部为热门头条的轮播展示，用户滑动可以切换信息内容。中部为本周的热门话题，用户可以在此进入参与话题互动；底部为头条的文章推送列表，以标题加图片的形势展示，点击可查看文章的详细内容，具体实现如下图所示：



图 6-11 头条首页图

#### 4) 我的模块页面

该模块为用户管理中心页面，主要提供两个功能，一是用户基本个人信息管理，二是用户社交互动信息管理。因此，本页面展示用户的昵称，自我介绍，头像，关注的用户，收藏，点赞的贴子。同时，提供用户发布贴子，以及发出的评论入口。用户可以在此编辑个人信息，设置昵称，与孩子的关系，所在地区等基本个人信息以及更改手机号码等私密信息。除了这两个功能之外，用户还可以在此查看孩子的成长记录，具体实现如下图所示：





图 6-12 个人信息首页



图 6-13 编辑个人信息界面



图 6-14 选择用户地区界面



图 6-15 删除帖子提示页面





图 6-16 设置手机号界面



图 6-17 查看发布评论界面

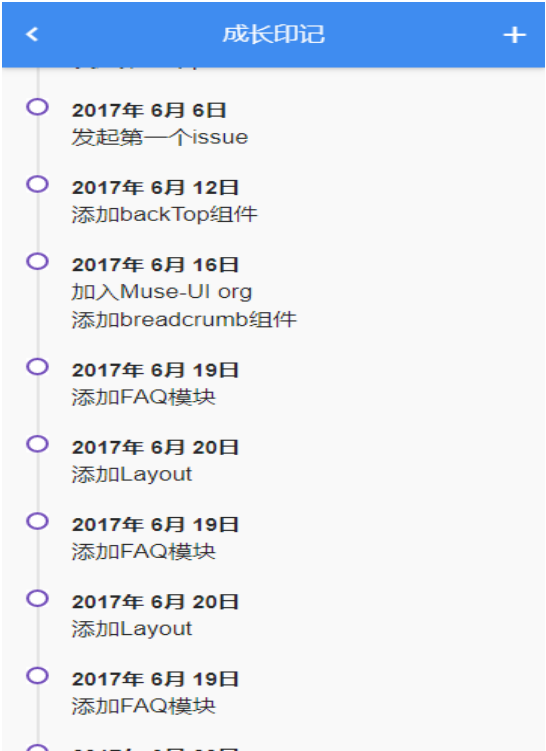


图 6-18 查看成长印记

## 6.5 部分功能代码分析

### (1) 项目入口文件代码

该入口文件位置在 parent\_help\src\main.js 下,主要作用是初始化 vue 实例,并且引入相关依赖库,具体代码如下所示:

```
1 //...
2
3 import Vue from 'vue'
4 import Router from 'vue-router'
5 import store from './vuex'
6 import Vuex from 'vuex'
7 import App from './App'
8 import routes from './router'
9 import axios from 'axios'
10 import ImgLazyLoad from 'vue-lazyload'
11 import MuseUi from 'muse-ui'
12 import 'muse-ui/dist/muse-ui.css'
13 import 'muse-ui/dist/theme-light.css'
14 import Toast from './components/Toast.js'
15 import './components/Toast.css'
16
17 Vue.use(Toast)
18 Vue.use(Router)
19 Vue.use(MuseUi)
20 Vue.config.productionTip = false
21
22 window.axios = axios;
23 Vue.use(ImgLazyLoad, {
24   loading: 'http://image.heitem.com/timg.gif',
25 })
26 axios.defaults.baseURL = 'https://www.easy-mock.com/mock/5a9559d7a563ca3b10c484b0/parent_help';
27 axios.defaults.autoFaceURL = 'https://www.easy-mock.com/mock/5a9559d7a563ca3b10c484b0/parent_help';
28
29
30 const router = new Router({
31   mode: 'history',
32   routes,
33 })
34
35
36 new Vue({
37   el: '#app',
38   router,
39   store,
40   render: h => h(App),
41   template: '<App/>',
42   components: { App }
43 })
```

图 6-19 项目入口图

### (2) 应用路由代码

应用的路由切换是通过 vue-router 实现,通过设置路径和组件,即可实现视图的切换。具体设置如下代码所示:

```

1  import Home      from '@page/home/Home'
2  import User      from '@page/user/index'
3  import Login     from '@page/user/login'
4  import Loginpassword from '@page/user/loginpassword'
5  import Power     from '@page/user/power'
6  import Register  from '@page/user/register'
7  import GrowthMark from '@page/user/growthMark'
8  import Publish   from '@page/publish/index'
9  import Atlas     from '@page/Exhibition/atlas'
10 import More      from '@page/more/index'
11 import keyword   from '@page/user/keyword'
12 import BlackList from '@page/user/blacklist'
13 import ModifyData from '@page/user/modifydata'
14 import CellPhone from '@page/user/cellphone'
15 import Follow    from '@page/user/follow'
16 import HomeIndex from '@page/index'
17 import UserIssUe from '@page/user/Userissue'
18 import DiscUss   from '@page/user/discuss'
19 import DiscoVer  from '@page/more/discover'
20 import MoreLaBel from '@page/more/label'
21 import InforMation from '@page/more/information'
22 import RanKing   from '@page/more/ranking'
23 import OthErs    from '@page/others/index'
24 import exhibition from '@page/Exhibition/home'
25 import Topic     from '@page/Exhibition/topic'
26
27 export default [
28   {
29     path: '/',
30     redirect: { path: '/home' },
31     component: Home,
32   },
33   {
34     path: '/index',
35     name: '首页四栏',
36     component: HomeIndex,
37     children: [
38       {path: '/home', name: '主页', component: Home},
39       {path: '/more', name: '更多', component: More},
40       {path: '/user', name: '个人中心', component: User},
41       {path: '/others/:name', name: '其他用户的资料', component: OthErs},
42       {path: '/exhibition', name: '展集', component: exhibition}
43     ]
44   }
]

```

图 6-20 应用路由配置图

### (3) 应用首页前端代码

前端页面展示通过第三方库 muse-ui 构建, muse-ui 拥有丰富的组件, 通过传入 props 的值和调用暴露的公共逻辑方法改变视图渲染, 使前端的页面能够模块化, 提高代码的质量。

```

1 <template>
2 <div>
3 <router-view></router-view>
4 <mu-popup position="bottom" popupClass="Login-popup-bottom" :open="LoginBottom" @close="LoginBottomClose">
5 <mu-content-block class="Login-Bottom">
6 <mu-flat-button label="关闭" icon="clear" class="Login-Bottom-Close" @click="LoginBottomClose" /><br />
7 <div class="Login-Bottom-Icon">
8 
9 </div>
10 <mu-sub-header style="...">请先登录应用</mu-sub-header>
11 <mu-sub-header style="...">输入11位进入登录页, 输入12位进入注册页!</mu-sub-header>
12 <mu-text-field iconClass="Phone-Input" v-model="PhoneNumber" type="number" icon="phone iphone" labelFloat fullWidth/>
13 <mu-raised-button label="下一步" @click="login" secondary fullWidth />
14 </mu-content-block>
15 </mu-popup>
16 <mu-snackbar v-if="toast" :message="indexToast" action="确定" @actionClick="hideToast" @close="hideToast"/>
17 <mu-paper class="Bottom-Nav-Class">
18 <mu-bottom-nav :value="bottomNav" @change="handleChange">
19 <mu-bottom-nav-item value="home" title="社区" icon="home" to="/home" />
20 <mu-bottom-nav-item value="exhibition" title="成长" icon="collections" to="/exhibition/atlas" />
21 <mu-float-button icon="add" @click="Publish" />
22 <mu-bottom-nav-item value="more" title="头条" icon="explore" to="exhibition"/>
23 <mu-bottom-nav-item value="user" title="我的" icon="account circle" />
24 </mu-bottom-nav>
25 </mu-paper>
26 <div class="seize-seat-bottom"></div>
27 </div>
28 </template>
29

```

图 6-21 首页前端代码

#### (4) 获取内容数据代码

应用的数据通过 axios 库发送 get 请求从后台服务器获得, 由于 axios 本质是一个 promise, 因此需要 then 方法内处理数据, 保存到全局 data 中, 供页面调用渲染调用。

```

313 LoginData () {
314   const active_Tab = this.activeTab;
315   if (active_Tab !== '') {
316     this.$loading(active_Tab_CN + '话题正在' + '载入中...');
317   }
318   axios.get(this.activeTab)
319     .then(res => {
320       if (res.status === 200) {
321         const Home_Api = this.activeTab;
322         if (Home_Api === 'index') {
323           this.mockData = res.data.data.list;
324         }
325       }
326       let self = this;
327       setTimeout(function () {
328         self.closeLoading()
329       }, 2000)
330     })
331 }

```

图 6-22 获取后台数据函数代码

### (5) vuex 状态管理代码

```
1 import Vue from 'vue'
2 import Vuex from 'vuex'
3
4 Vue.use(Vuex)
5
6 export default new Vuex.Store({
7   state: {
8     UserPhone: '',
9   },
10  mutations: {
11    LoginPhone (state, phone) {
12      state.UserPhone = phone;
13      sessionStorage.setItem('userphone', JSON.stringify(state.UserPhone))
14    }
15  },
16 })
```

图 6-23 vuex 数据管理代码

## 第 7 章 系统测试

### 7.1 单元测试

#### 7.1.1 导航切换

表 7-1 底部导航切换用例表

用例编号	ZWFW-DYCS-01	模块名称	底部导航
用例作者	廖艺宇	审核结果	通过
测试类型	功能		
用例描述	导航切换		
前置条件	用户已经通过账号密码登录应用		
(续上表)			
操作步骤	用户打开应用，点击底部导航切换内容		
输入数据	用户的账号，密码		
预期结果	点击不同导航，切换到不同的模块		
实际结果	如预期结果		
结论	测试通过		

#### 7.1.2 社区首页测试

表 7-2 社区首页测试用例表

用例编号	ZWFW-DYCS-02	模块名称	社区首页
用例作者	廖艺宇	审核结果	通过
测试类型	功能		
用例描述	查看社区首页		
前置条件	从导航处，点击选择进入社区模块		
操作步骤	①进入查看社区中贴子的内容展示， ②顶部切换社区内容分类 ③		

(续上表)

输入数据	
预期结果	
实际结果	如预期结果
结论	测试通过

7.1.3 个人信息测试

表 7-2 个人信息测试用例表

用例编号	ZWFW-DYCS-02	模块名称	个人信息
用例作者	廖艺宇	审核结果	通过
测试类型	功能		
用例描述	查看个人信息中心		
前置条件	用户已经通过账号密码成功登录应用		
操作步骤	①用户打开应用，点击进入个人信息模块，依次查看已发布的贴子，已发布的评论，已收藏的贴子，进入孩子的成长历程。 ②修改个人信息相关个人信息资料 ③退出登录 ④修改个人信息，退出登录，查看最新资料		
输入数据			
预期结果	所有页面能够成功切换， 修改个人信息成功，重新登陆后，显示信息为修改后的信息		
实际结果	如预期结果		
结论	测试通过		

7.2 功能测试

7.2.1 发布贴子功能测试

表 7-3 发布贴子功能测试用例表

用例编号	ZWFW-GNCS-01	模块名称	注册
用例作者	廖艺宇	审核结果	通过
测试类型	功能		
用例描述	检查发布贴子功能		

(续上表)

前置条件	用户已经通过账号密码登录应用
操作步骤	用户打开应用，点击底部导航栏处的发布按钮 进入发布贴子内容填写模块 填写完成，点击发布
输入数据	贴子的标题，内容，类型相关字段
预期结果	用户填写的内容如有敏感词语，系统应用提示用户重新编辑 点击发布成功，后台新增数据到贴子内容表
实际结果	如预期结果
结论	测试通过

7.2.2 系统提示登录功能测试

表 7-4 系统提示登录功能测试用例表

用例编号	ZWFW-GNCS-01	模块名称	注册
用例作者	廖艺宇	审核结果	通过
测试类型	功能		
用例描述	检查注册功能		
前置条件	用户尚未登录或登录缓存已失效		
操作步骤	用户进入应用，选择评论贴子，点击发贴功能或进入用户管理中心模块		
输入数据	无		
预期结果	系统应用弹出登录提示窗口		
实际结果	如预期结果		
结论	测试通过		

7.2.3 注册功能测试

表 7-5 注册功能测试用例表

用例编号	ZWFW-GNCS-01	模块名称	注册
用例作者	廖艺宇	审核结果	通过
测试类型	功能		
用例描述	检查注册功能		
前置条件	游客尚未注册成为用户		



(续上表)

操作步骤	游客打开应用，点击进入个人信息模块， 系统提示用户需要登录， 游客选择注册，输入相关手机号码 点击确认注册
输入数据	
预期结果	游客能够通过手机号码接收验证码， 前端能够发送验证码给后台 后台检测验证码是否正确， 注册成功，并把用户数据保存到后台数据库
实际结果	如预期结果
结论	测试通过

## 第 8 章 总结与展望

### 8.1 工作总结

历经几个月的时间，毕业设计最终落入尾声，期间遇到了很多难题和技术壁垒，深深感受到自身知识的短缺，唯有不断学习和思考，在巩固所学知识之外，增强知识储备，不断探索新的知识。

在本次毕业设计过程中，给到我最大的感受就是应用的市场定位，在功能设计上面花了不少时间去思考，究竟目前家长需要一个怎样的社交平台去推动家庭教育的发展，才能让孩子能够健康快乐地成长呢？在这个问题还没得到解决之前，一直未能去通过代码去实现具体功能。我想对于目前很多科技创业公司来说也是同样的道理，在一个产品设计中，如果只是盲目跟从技术潮流，找不到合适的市场定位，我想这个产品最终很大可能会夭折。

我离学校出来实习已经半年多，半年来由于都是接触到公司底层框架，很多思维已经形成了一种习惯，毕业设计其实对于自己来说，正是一个很好的机会去探索新的知识，可以认识到自己那方面存在的不足，缺点，以后加以改进，我想对自己以后的软件开发道路有很好的帮助。

### 8.2 不足与展望

由于自己一直从事前端开发工作，对后台数据库的设计理解不深，加以能力有限，时间有限，知识储备不足，未能完成当初应用设计的全部功能，实在遗憾。本人觉得还可以从一下方面进行扩展：

- ①补充IM消息聊天系统互动
- ②增加家长育儿计划
- ③建立家长课堂

这几个方面需要花更多的时间和技术储备去完成，我也将不断地维护更新应用，不断积累知识，在未来的生活道路不怕困难，刻苦专研。

## 参考文献

- [1] 胡月, 王以宁, 徐鹏, 张海. 互联网时代教育类 APP 开发——美国《教育技术开发人员指南》手册解读及启示[J]. 中国电化教育.
- [2] 王畅. 论家庭教育的重要性[J]. 文学教育(下).
- [3] 袁蓉蓉. Web2.0 环境下小学班级主题学习社区的构建与应用研究[D]. 西南大学 2009
- [4] 莫昌星, 李磊, 郭湘南, 余道敏, 肖伟明. 基于 Android 的智慧社区综合运营系统的设计[J]. 电子设计工程. 2016(09)
- [5] 周萌, 姜增辉, 吴际达, 吴海军. 响应式快速开发框架研究与实现[J]. 电子科学技术. 2016(04)
- [6] 佟伟光, 郭霏霏. 软件测试[M]. 北京:人民邮电出版社, 2015. 1: 73-98
- [7] 丁永贤. 高校移动校园 APP 的设计与分析[J]. 数字通信世界. 2017(10)
- [8] 朱二华. 基于 Vue.js 的 Web 前端应用研究[J]. 科技与创新. 2017(20)
- [9] 李颖. 基于响应式设计的 Web App 界面布局研究[J]. 安庆师范大学学报(自然科学版). 2017(03)
- [10] 周晶. 浅谈手机 UI 界面的人性化设计[J]. 电脑迷. 2016(07)
- [11] 衣杨. 基于面向对象的软件分析与设计实验教程[M]. 广州:中山大学出版社, 2011. 3:1-3

## 致 谢

首先，非常感受朱荣导师的悉心指导，由于自己在外实习，给予了我很大的支持和理解，使我能够完成大学生涯中最后一门课程。同时感谢身边所有的同学和朋友，让我在大学四年留下了很多珍贵的回忆，祝愿各位前程似锦。

谢谢！

