

技术栈报告

第一组

2020-04-01

1 项目简述

本项目旨在开发一个新的校内论坛，供师生在论坛网站上进行交流讨论。囊括以下基本功能：发帖帖帖、上传音频片件、以及关注用户等。

2 前端框架

2.1 React

当前最流行的几个前端框架如React、Vue、Angular和Bootstrap。但如今大多数公司还是选择的React作为前端框架，且有团队对其维护，并且社区生态较为完善，所以对于没有接触过React框架的新人较为友好，简单易学，因此我们采用该前端框架。React的设计思想极其独特，属于革命性创新，性能出众，代码逻辑却非常简单。所以，越来越多的人开始关注和使用，认为它可能是将来Web 开发的主流工具。这个项目本身也越滚越大，从最早的UI引擎变成了一整套前后端通吃的Web App 解决方案。衍生的React Native 项目，目标更是宏伟，希望用写Web App 的方式去写Native App。如果能够实现，整个互联网行业都会被颠覆，因为同一组人只需要写一次UI，就能同时运行在服务器、浏览器和手机。

2.2 HTML5

定义网页的内容。HTML5是构建Web内容的一种语言描述方式。HTML5是互联网的下一代标准，是构建以及呈现互联网内容的一种语言方式，被认为是互联网的核心技术之一。HTML产生于1990年，1997年HTML4成为互联网标准，并广泛应用于互联网应用的开发。

2.3 CSS

规定网页的布局。层叠样式表(Cascading Style Sheets)是一种用来表现HTML（标准通用标记语言的一个应用）或XML（标准通用标记语言的一个子集）等文件样式的计算机语言。CSS不仅可以静态地修饰网页，还可以配合各种脚本语言动态地对网页各元素进行格式化。

2.4 JavaScript

对网页行为进行编程JavaScript是一种具有函数优先的轻量级，解释型或即时编译型的编程语言。虽然它是作为开发Web页面的脚本语言而出名的，但是它也被用到了很多非浏览器环境中，JavaScript 基于原型编程、多范式的动态脚本语言，并且支持面向对象、命令式和声明式（如函数式编程）风格。

3 后端框架

3.1 Spring Boot

1. 简介:

Spring Boot是由Pivotal团队提供的全新框架，其设计目的是用来简化新Spring应用的初始搭建以及开发过程。该框架使用了特定的方式来进行配置，从而使开发人员不再需要定义样板化的配置。

2. 优势:

- 快速创建独立运行的Spring项目以及与主流框架集成
- 使用嵌入式的Servlet容器，应用无需打成WAR包
- Starters自动依赖与版本控制
- 大量的自动配置，简化开发，也可修改默认值
- 无需配置XML，无代码生成，开箱即用
- 准生产环境的运行时应用监控

3. 核心特点:

- 微服务：使用Spring Boot可以生成独立的微服务功能单元
- 自动配置：针对很多Spring应用程序常见的应用功能，Spring Boot能自动提供相关配置
- 起步依赖：告诉Spring Boot需要什么功能，它就能引入需要的库
- 命令行界面：这是Spring Boot的可选特性，借此你只需写代码就能完成完整的应用程序，无需传统项目构建
- Actuator：帮助你监控和管理Spring Boot应用

3.2 MySQL

由于市面上的关系型数据库大多数收费，且我们项目不是特别大，因此MySQL是一个比较好的选择。MySQL是一个关系型数据库管理系统，由瑞典MySQL AB 公司开发，属于Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一，在WEB 应用方面，MySQL是最好的RDBMS (关系数据库管理系统) 应用软件之一。MySQL是一种关系型数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

4 项目管理工具

4.1 项目生命周期管理工具——Redmine

基于web，提供项目管理和缺陷跟踪功能。以WEB形式把成员、任务、文档、讨论及各种形式的资源组织在一起，推动项目的进度，可对接GIT、SVN等版本管理系统

- Download:

官网: <https://www.redmine.org/projects/redmine/wiki/Download>

一键安装: <https://bitnami.com/stack/redmine>

Git: <https://github.com/redmine/redmine>

- 优点:

- 1.开源

- 2.功能完善，可定制，有日历，可以自动计算生成Gantt chart

- 3.相对灵活轻量，适合小团队

- Warning:

- 1.和各种商业项目管理工具比起来有点陈旧，存在部分免费插件年久失修

- 2.追踪流程变更的时候比较麻烦，最好规划好再录入

- 3.丑，但是别的开源更丑

4.2 版本控制工具——Github

GitHub是当前最受欢迎的版本控制工具。GitHub是一个面向开源及私有软件项目的托管平台，因为只支持git作为唯一的版本库格式进行托管，故名GitHub。GitHub于2008年4月10日正式上线，除了Git代码仓库托管及基本的Web管理界面以外，还提供了订阅、讨论组、文本渲染、在线文件编辑器、协作图谱（报表）、代码片段分享（Gist）等功能。目前，其注册用户已经超过350万，托管版本数量也是非常之多，其中不乏知名开源项目Ruby on Rails、jQuery、python等。

4.3 持续集成和发布工具——Gitlab-CI

- 简介:

Gitlab-CI是Gitlab官方提供的持续集成服务，我们可以在仓库的根目录下新建.gitlab-ci.yml文件，自己定义持续集成流程模板，并且在Gitlab中配置runner，在之后的每次提交合并中将会触发构建，并且可以通过Gitlab的hook，在代码提交的各个环节自动地完成一系列的构建工作。

- 作用:

完成软件的自动化工作（如打包，单元测试，部署等），提高工作效率

- 优点:

- 1.CSDN上有许多参考文档
- 2.与版本控制工具配合良好
- 3.提供对Docker的支持
- 4.提供免费版本

4.4 编译工具——WebStorm、IntelliJ IDEA

WebStorm是jetbrains公司旗下一款JavaScript 开发工具。已经被广大中国JS开发者誉为“Web前端开发神器”、“最强大的HTML5编辑器”、“最智能的JavaScript IDE”等。与IntelliJ IDEA同源，继承了IntelliJ IDEA强大的JS部分的功能。IntelliJ IDEA是java编程语言开发的集成环境。IntelliJ在业界被公认为最好的java开发工具，尤其在智能代码助手、代码自动提示、重构、J2EE支持、各类版本工具(git、svn等)、JUnit、CVS整合、代码分析、创新的GUI设计等方面的功能可以说是超常的。

4.5 测试工具——JMeter

- 原理:

JMeter是一个桌面应用，被设计为用于测试c/s结构的应用程序，如web应用或FTP应用。是到目前为止，一个使用得最广泛的开源测试工具。它是由纯java编写的，通过其API可以对它进行扩展。JMeter位于c/s中的客户端工作，它可以测试服务器的响应时间，CPU负载，内存占用和资源的使用。绝大多数的JMeter测试脚本或测试计划都涉及到创建一个循环，按照预先定义的时间间隔和线程组来模拟向服务器发出请求。这些测试脚本可以定制和扩展。

- 作用:

1. 能够对HTTP和FTP服务器进行压力和性能测试，也可以对任何数据库进行同样的测试（通过JDBC）
2. 完全的可移植性和100% 纯java
3. 完全多线程框架允许通过多个线程并发取样和通过单独的线程组对不同的功能同时取样
4. 缓存和离线分析/回放测试结果

- 优点:

1. 零购置成本——可直接从官网下载得到

2. 学习曲线平滑——只需要基本的html和正则表达式知识
3. 功能全面——不仅只能测试web应用
4. 架构可伸缩——其组件被设计为可通过合并完成大规模测试。测试可以在多台主机上同时运行
5. API可扩展——API开放，并可由用户扩展编程
6. 技术支持良好——包括在线用户指南、用户论坛、网络文章

4.6 docker-composer

1. 简介:

Docker 是一个开源的应用容器引擎，基于Go 语言并遵从Apache2.0 协议开源。Docker 可以让开发者打包他们的应用以及依赖包到一个轻量级、可移植的容器中，然后发布到任何流行的Linux 机器上，也可以实现虚拟化。

Docker-Composer是docker 提供的一个命令行工具，用来定义和运行由多个容器组成的应用。使用composer，我们可以通过YAML 文件声明式的定义应用程序的各个服务，并由单个命令完成应用的创建和启动。

2. 优势:

- 更高效的利用系统资源
- 持续支付和部署
- 更轻松的维护和拓展
- 更轻松的迁移

3. 核心特点:

- 轻量级：单机可以轻松支持上百Container，让各种个位数虚拟化的方案相形见绌
- 标准：Docker 容器基于开放式标准，能够在所有主流Linux 版本、Microsoft Windows 以及包括VM、裸机服务器和云在内的任何基础服务设施上运行
- 快速就绪：一秒以内启动，即使是以资源快速就绪著称的青云IAAS也无法相比
- 弱安全：Docker能够对多种OS资源进行隔离

4.7 maven

- 简介:

Maven 翻译为“专家”、“内行”，是Apache 下的一个纯Java 开发的开源项目。基于项目对象模型（POM）概念，Maven利用一个中央信息片断能管理一个项目的构建、报告和文档等步骤。

Maven 是一个项目管理工具，可以对Java 项目进行构建、依赖管理。

Maven 也可被用于构建和管理各种项目，例如C, Ruby, Scala 和其他语言编写的项目。Maven 曾是Jakarta 项目的子项目，现为由Apache 软件基金会主持的独立Apache 项目。

Maven能够帮助开发者完成构建、文档生成、报告、依赖、SCMs、发布、分发、邮件列表等功能。

- 优势:

- 使用Maven管理的Java项目都有相同的结构
- 统一维护jar包
- 自动进行依赖管理

- 核心特点:

- 项目设置遵循统一的规则
- 任意工程中共享
- 依赖管理包括自动更新
- 一个庞大且不断增长的库
- 可扩展，能够轻松编写Java 或脚本语言的插件
- 只需很少或不需要额外配置即可即时访问新功能
- 基于模型的构建Maven能够将任意数量的项目构建到预定义的输出类型中，如JAR, WAR 或基于项目元数据的分发，而不需要在大多数情况下执行任何脚本
- 项目信息的一致性站点使用与构建过程相同的元数据，Maven 能够生成一个网站或PDF，包括要添加的任何文档，并添加到关于项目开发状态的标准报告中
- 发布管理和发布单独的输出Maven 将不需要额外的配置，就可以与源代码管理系统（如Subversion 或Git）集成，并可以基于某个标签管理项目的发布。它也可以将其发布到分发位置供其他项目使用。Maven 能够发布单独的输出，如JAR，包含其他依赖和文档的归档，或者作为源代码发布。
- 向后兼容性可以很轻松的从旧版本Maven 的多个模块移植到Maven 3 中

- 子项目使用父项目依赖时，正常情况子项目应该继承父项目依赖，无需使用版本号
- 并行构建编译的速度能普遍提高20 - 50
- 更好的错误报告Maven 改进了错误报告，提供了Maven wiki 页面的链接，可以通过点击链接查看错误的完整描述

4.8 redux

1. 简介:

Redux对于JavaScript应用而言是一个可预测状态的容器，即一个应用数据流框架。最主要是用作应用状态的管理，它用一个单独的常量状态树（对象）保存这一整个应用的状态，这个对象不能直接被改变。当一些数据变化了，一个新的对象就会被创建（使用actions和reducers）。

2. 优势:

- 始终有一个准确的数据源，就是store, 对于如何将actions以及应用的其他部分和当前的状态同步可以做到绝不混乱
- 具备可预测结果的性质和严格的组织结构让代码更容易维护
- 对代码应该如何组织更加严苛，这使代码更加一致，对团队协作更加容易
- 编写可测试代码的首要准则就是编写可以仅做一件事并且独立的小函数.Redux的代码几乎全部都是这样的函数：短小、纯粹、分离
- 可以带来更好的用户体验并且有助于搜索引擎优化，尤其是对于首次渲染。仅仅是把服务端创建的store传递给客户端就可以
- 开发者可以实时跟踪在应用中正在发生的一切，从actions到状态的改变
- 存在很多支持Redux的社区，使它能够吸引更多的人来使用

3. 核心特点:

Redux核心特点有三个:

- Actions:

简单地，Actions就是事件。Actions传递来自这个应用（用户接口，内部事件比如API调用和表单提交）的数据给store。store只获取来自Actions的信息。内部Actions就是简单的具有一个type属性（通常是常量）的JavaScript对象，这个对象描述了action的类型以及传递给store的负载信息。

Actions通过action生成器创建，它们仅仅是返回action的函数。

要在应用中的任何地方调用actions很容易，使用dispatch方法。

- Store:

Store对象保存应用的状态并提供一些帮助方法来存取状态，分发状态以及注册监听。全部state由一个store来表示。任何action通过reducer返回一个新的状态对象。这就使得Redux非常简单以及可预测。

- Reducers:

在函数式JavaScript中reducer基于数组reduce方法，接收一个回调（reducer）让你从多个值中获得单个值，整数和，或者一个一系列值的累积。在Redux中，reducer就是获得这个应用的当前状态和事件然后返回一个新状态的函数。理解reducer是怎样工作的至关重要，因为它们完成大部分工作。这是一个非常简单的reducer，通过获取当前state和一个action作为参数，再返回下一个state。

对于更多复杂的项目，使用Redux提供的combineReducers()实例是必要的（推荐）。它把在这个应用中所有的reducer结合在一起成为单个索引reducer。每一个reducer负责它自己那部分应用的状态，这个状态参数和其他reducer的不一样。combineReducers()实例使文件结构更容易维护。

如果一个对象（state）只改变一些值，Redux就创建一个新的对象，那些没有改变的值将会指向旧的对象而且新的值将会被创建。这对性能是极好的。为了让它更有效率你可以添加Immutable.js

4.9 nginx

Nginx (engine x) 是一个高性能的HTTP和反向代理web服务器，同时也提供了IMAP/POP3/SMTP服务。Nginx是由伊戈尔·赛索耶夫为俄罗斯访问量第二的Rambler.ru站点（俄文：Рамблер）开发的，第一个公开版本0.1.0发布于2004年10月4日。

其将源代码以类BSD许可证的形式发布，因它的稳定性、丰富的功能集、示例配置文件和低系统资源的消耗而闻名。2011年6月1日，nginx 1.0.4发布。

Nginx是一款轻量级的Web 服务器/反向代理服务器及电子邮件（IMAP/POP3）代理服务器，在BSD-like 协议下发行。其特点是占有内存少，并发能力强，事实上nginx的并发能力在同类型的网页服务器中表现较好，中国大陆使用nginx网站用户有：百度、京东、新浪、网易、腾讯、淘宝等。