****

**本 科 毕 业 设 计**

|  |  |
| --- | --- |
| **题**  **目** | 基于Node.js的Web 框架自动生成工具 |

**作 者：** 韦 远

**专 业：** 计算机科学与技术

**指导教师：** 朱 晓 辉

**完成日期：** 2017年 5 月 31日

摘要

目录

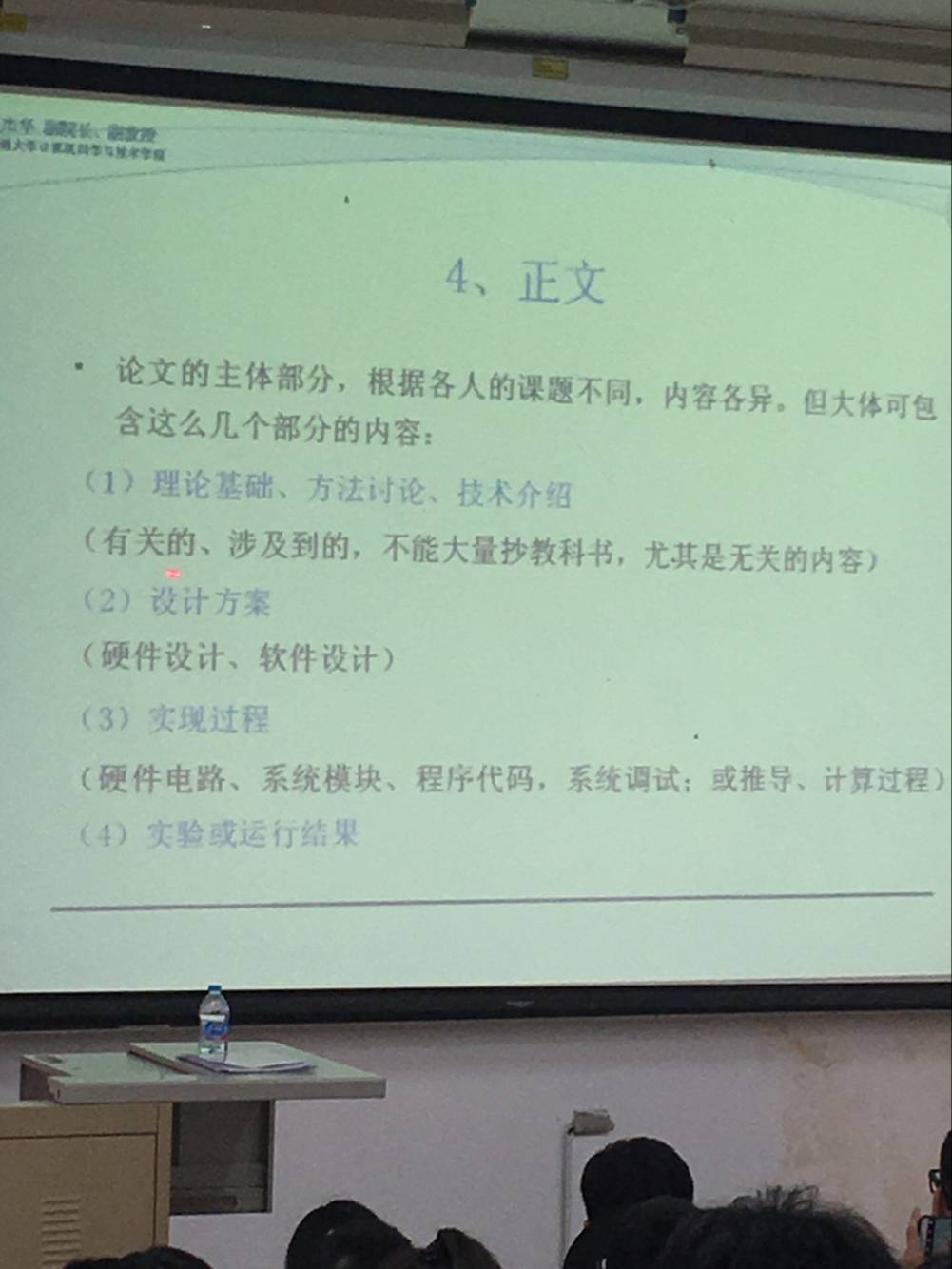
引言

正文

参考文献

致谢

附录



# 引言

随着互联网技术的不断发展，网站已经成为人们不可或缺的一部分，人们无时不刻不在浏览各式各样的网站，而网站也经历了从初始的静态页面到现在的充满大量的用户交互的过程。程序员在这里就起到了很重要的作用，是他们将网站搭建出来，但是随着Web技术的不断发展，各式各样的Web前端与后端的技术出现在了市场上面，前端有angularjs，vue，react等，后端有nodejs，php，java等，让程序员不知道该如何选择。所以为了网站快速而高效的开发，我们创建了一个以angularjs为前端框架，nodejs为后端支撑，mongodb为数据支持的网站模板，再配合webpack强大的打包功能，以及前后端统一使用JavaScript语言开发，使用我们的这个模板来开发网站将体会到前所未有的快速轻量与高效。

# 正文

## 技术介绍

1. Node.js

Node.js是一个Javascript运行环境(runtime)。实际上它是对Google V8引擎进行了封装。V8引 擎执行Javascript的速度非常快，性能非常好。Node.js对一些特殊用例进行了优化，提供了替代的API，使得V8在非浏览器环境下运行得更好。

Node.js是一个基于Chrome JavaScript运行时建立的平台， 用于方便地搭建响应速度快、易于扩展的网络应用。Node.js 使用事件驱动， 非阻塞I/O 模型而得以轻量和高效，非常适合在分布式设备上运行数据密集型的实时应用。

V8引擎本身使用了一些最新的编译技术。这使得用Javascript这类脚本语言编写出来的代码运行速度获得了极大提升，又节省了开发成本。对性能的苛求是Node的一个关键因素。 Javascript是一个事件驱动语言，Node利用了这个优点，编写出可扩展性高的服务器。Node采用了一个称为“事件循环(event loop）”的架构，使得编写可扩展性高的服务器变得既容易又安全。提高服务器性能的技巧有多种多样。Node选择了一种既能提高性能，又能减低开发复杂度的架构。这是一个非常重要的特性。并发编程通常很复杂且布满地雷。Node绕过了这些，但仍提供很好的性能。

Node采用一系列“非阻塞”库来支持事件循环的方式。本质上就是为文件系统、数据库之类的资源提供接口。向文件系统发送一个请求时，无需等待硬盘（寻址并检索文件），硬盘准备好的时候非阻塞接口会通知Node。该模型以可扩展的方式简化了对慢资源的访问， 直观，易懂。尤其是对于熟悉onmouseover、onclick等DOM事件的用户，更有一种似曾相识的感觉。

虽然让Javascript运行于服务器端不是Node的独特之处，但却是其一强大功能。不得不承认，浏览器环境限制了我们选择编程语言的自由。任何服务器与日益复杂的浏览器客户端应用程序间共享代码的愿望只能通过Javascript来实现。虽然还存在其他一些支持Javascript在服务器端 运行的平台，但因为上述特性，Node发展迅猛，成为事实上的平台。

在Node启动的很短时间内，社区就已经贡献了大量的扩展库（模块）。其中很多是连接数据库或是其他软件的驱动，但还有很多是凭他们的实力制作出来的非常有用的软件。

最后，不得不提到的是Node社区。虽然Node项目还非常年轻，但很少看到对一个项目如此狂热的社区。不管是新手，还是专家，大家都围绕着项目，使用并贡献自己的能力，致力于打造一个探索、支持、分享、听取建议的乐土。

Node.js的优点

nodejs作为一个新兴的前端框架，后台语言，有很多吸引人的地方：

RESTful API

单线程

Node.js可以在不新增额外线程的情况下，依然可以对任务进行并发处理 —— Node.js是单线程的。它通过事件轮询（event loop）来实现并发操作，对此，我们应该要充分利用这一点 —— 尽可能的避免阻塞操作，取而代之，多使用非阻塞操作。

非阻塞IO

V8虚拟机

事件驱动

1. Angularjs

AngularJS诞生于2009年，由Misko Hevery 等人创建，后为Google所收购。是一款优秀的前端JS框架，已经被用于Google的多款产品当中。AngularJS有着诸多特性，最为核心的是：MVC、模块化、自动化双向数据绑定、语义化标签、依赖注入等等。

AngularJS是为了克服HTML在构建应用上的不足而设计的。HTML是一门很好的伪静态文本展示设计的声明式语言，但要构建WEB应用的话它就显得乏力了。所以我做了一些工作（你也可以觉得是小花招）来让浏览器做我想要的事。

通常，我们是通过以下技术来解决静态网页技术在构建动态应用上的不足：

类库 - 类库是一些函数的集合，它能帮助你写WEB应用。起主导作用的是你的代码，由你来决定何时使用类库。类库有：jQuery等

框架 - 框架是一种特殊的、已经实现了的WEB应用，你只需要对它填充具体的业务逻辑。这里框架是起主导作用的，由它来根据具体的应用逻辑来调用你的代码。框架有：knockout、sproutcore等。

AngularJS使用了不同的方法，它尝试去补足HTML本身在构建应用方面的缺陷。AngularJS通过使用我们称为指令(directives)的结构，让浏览器能够识别新的语法。例如：

使用双大括号{{}}语法进行数据绑定；

使用DOM控制结构来实现迭代或者隐藏DOM片段；

支持表单和表单的验证；

能将逻辑代码关联到相关的DOM元素上；

能将HTML分组成可重用的组件。

1. Mongodb

MongoDB是一个基于分布式文件存储的数据库。由C++语言编写。旨在为WEB应用提供可扩展的高性能数据存储解决方案。

MongoDB是一个介于关系数据库和非关系数据库之间的产品，是非关系数据库当中功能最丰富，最像关系数据库的。他支持的数据结构非常松散，是类似json的bson格式，因此可以存储比较复杂的数据类型。Mongo最大的特点是他支持的查询语言非常强大，其语法有点类似于面向对象的查询语言，几乎可以实现类似关系数据库单表查询的绝大部分功能，而且还支持对数据建立索引。

它的特点是高性能、易部署、易使用，存储数据非常方便。主要功能特性有：

\*面向集合存储，易存储对象类型的数据。

mongodb集群参考

mongodb集群参考

\*模式自由。

\*支持动态查询。

\*支持完全索引，包含内部对象。

\*支持查询。

\*支持复制和故障恢复。

\*使用高效的二进制数据存储，包括大型对象（如视频等）。

\*自动处理碎片，以支持云计算层次的扩展性。

\*支持RUBY，PYTHON，JAVA，C++，PHP，C#等多种语言。

\*文件存储格式为BSON（一种JSON的扩展）。

\*可通过网络访问。

## 软件设计

### 需求分析

### 设计思路

现如今，web开发技术种类繁多，快速高效的开发乃是每个企业追求的目标，我从web开发效率的出发点上面考虑，集合多种web开发前沿的技术设计了这一套web开发的框架。

### 总体设计

### 详细设计

## 系统展示