



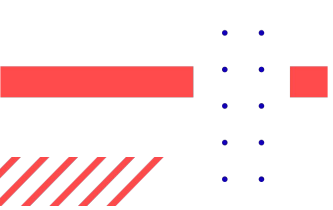
Web开发进阶班第1期

Lesson 1 Introduction & Web Technology

2023.07.15

www.jiangren.com.au

1. Introduction
2. 项目类型和难度
3. 课程要求(作业和项目)
4. Web基础知识介绍
5. 开发流程介绍



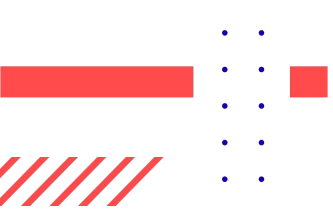
1. Introduction



课程介绍

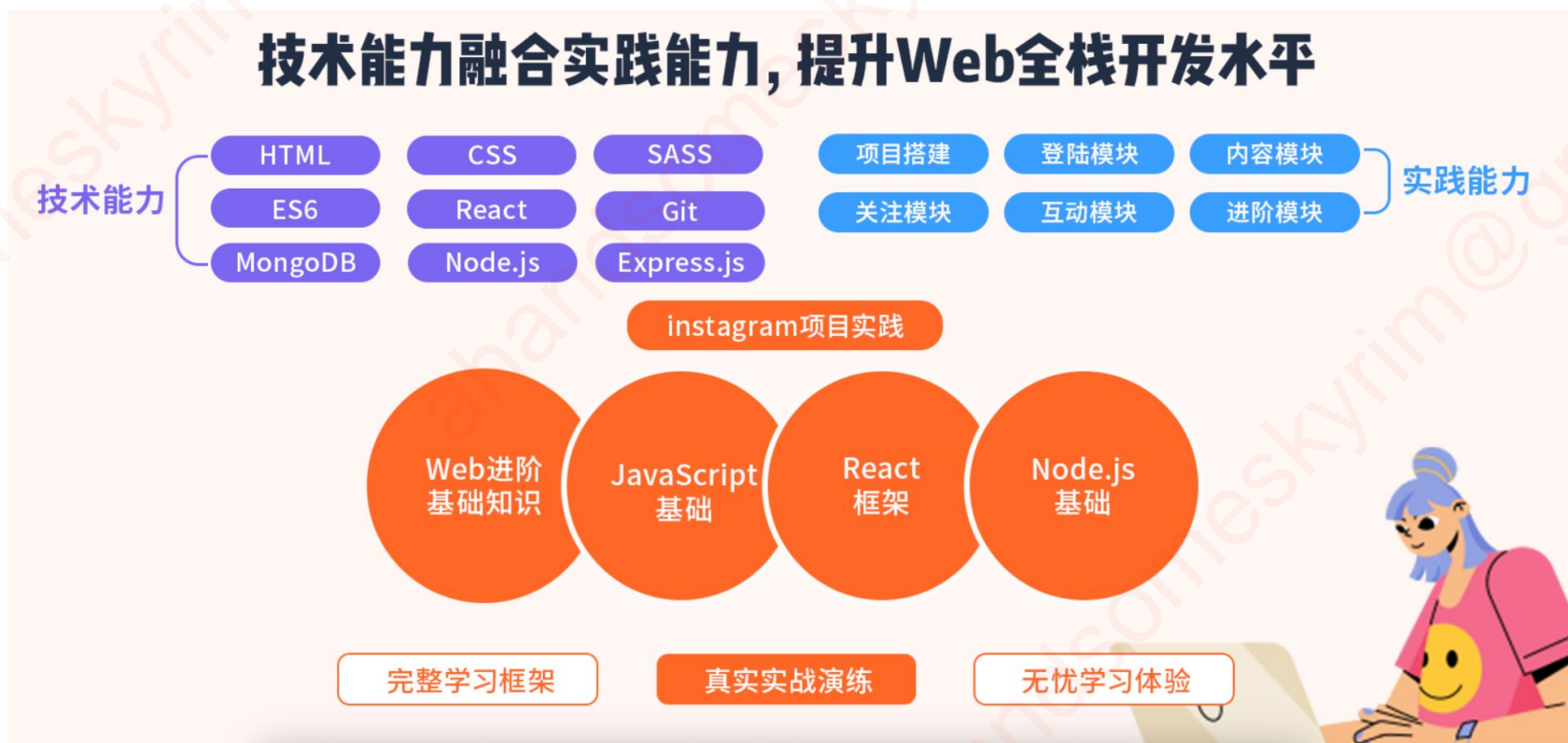
Web开发进阶班总共包含36节课程内容, 历时3个月, 共计约97小时, 通过直播形式授课。

进阶班的课程不仅能够帮助学生学习和巩固HTML, CSS, SASS, Git等Web开发基础技能, 夯实JavaScript基础, 熟悉React框架, 完善Node.js等基础。还重点融入了项目开发内容, 老师手把手带做项目, 从0-1搭建类Instagram APP, 以练代学, 提升上手实战能力, 基本达到入行就业水准。同时, 也为之后Web全栈班的学习打下坚实的基础。



1. Introduction

课程介绍



1. Introduction

课程介绍

为什么选择匠人的Web进阶开发班

我们的课程

理论+实战 助力求职

由浅入深、循序渐进的课程内容
帮助学生建立知识体系

老师手把手带教
及时解决学习过程中遇到的问题

课程融合项目，以练代学
注重培养学生动手开发能力

一线大厂大牛授课，同时配备随堂助教
课程跟踪班主任，全程跟踪学习情况及进度

庞大匠人校友资源
互相内推获取工作机会

全程参与项目开发落地
可以直接放在简历中

依靠项目指导，熟悉项目开发细节
在面试中脱颖而出

市面上的课程

实战机会少，教学水平弱

知识模块分散

讲解进度快
基础掌握不扎实

缺少上手能力

单一老师授课

缺少校友资源

缺少项目经历

缺少实践经验

自学

时间成本高，学习质量差

知识点分散
学习效率低

遇到问题难以解决

缺少实践经验

缺少导师资源

缺少校友资源

缺少项目经历

缺少实践经历

VS

VS

1. Introduction

课程介绍



Web入门班

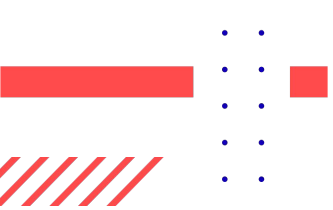
- **适合人群**
0基础学生
转专业人群
- **课程特点**
扎实基础能力
- **授课方式**
低强度学习网页开发技能 (html, css, javascript), SQL, Python)
- **项目实战**
老师带着手把手做成一个个人博客系统

Web开发进阶班

- **适合人群**
转专业人群
入门班学员
- **课程特点**
以练代学
老师带着做项目
全职IT辅导员
- **授课方式**
带领完成全栈项目
- **项目实战**
完成类Instagram App, 内容包含前后端开发, 数据库建立与链接

Web全栈班

- **适合人群**
IT专业学生
Web开发进阶班学生
- **课程特点**
多级别商业项目
面向企业求职
- **授课方式**
导师, tutor, 项目tutor, IT辅导员, 班主任多维度辅助学习, 完成知识+商业项目的高强度学习
- **项目实战**
全栈学员与DevOps班, BA实习组成团队项目组, 在Tutor带领下, 完成一个中型公司企业级项目。利用AWS, Kubernetes等实现云端多环境搭建, 面向企业面试官社招岗位需求



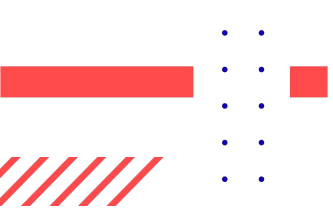
1. Introduction



导师介绍

大漠穷秋

15年的IT从业经验，参与或领导过公司多个核心业务系统和前端框架的研发工作，个人技术全面中兴软创前端技术架构师。著有《Ext江湖》、《ActionScript3.0游戏基础(第二版)》、《使用AngularJS开发下一代WEB应用》、《迈向Angular2》。专注前端领域多年，持有阿里云 ACP 认证，腾讯云 TVP 技术专家。Google Developers Experts协助Angular在中国的普及推广工作。



1. Introduction



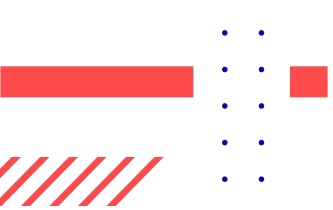
导师介绍

Kevin Luo

拥有多年的Web领域开发经验, T型软件工程师, AWS证书 x 3, 在程序设计开发和项目管理都有丰富的实践经历。

Ally Tang

Senior Full-Stack Developer, 曾为数学老师, 后转码上岸。拥有Azure solution architect expert和 Aws sap证书。目前从事前端、后端和 DevOps 开发, 在公司团队中负责项目部署和发布。



1. Introduction



学员介绍

目前的状态(转码, 全职工作, 学生)

遇到的瓶颈, 为什么报名课程

学完课程后希望达到的目标

2. 项目类型以及难度

个人主页
用户认证
实现密码重置、电子邮件验证
点赞、关注和取关
关注列表和被关注列表
消息、通知、搜索、SSO ...

从0-1实现 类Instagram App

学练结合
从项目出发强化开发能力

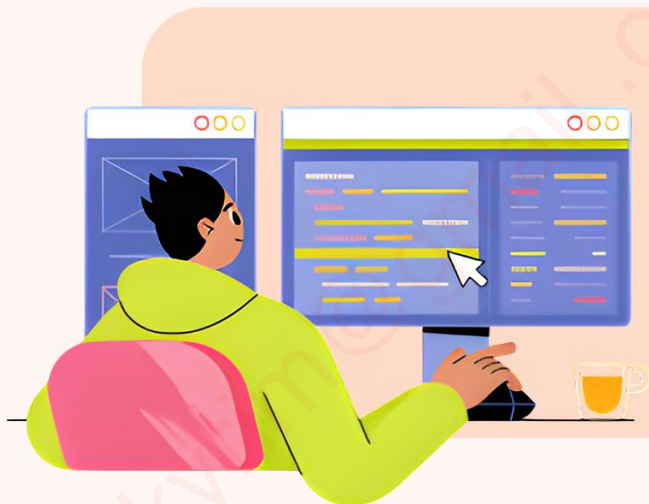
登陆注册

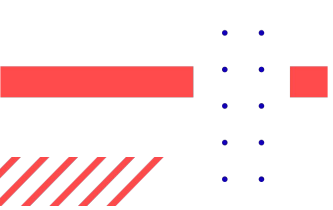
内容功能

互动功能

关注功能

进阶功能





2. 项目类型以及难度

Instagram

Sign In

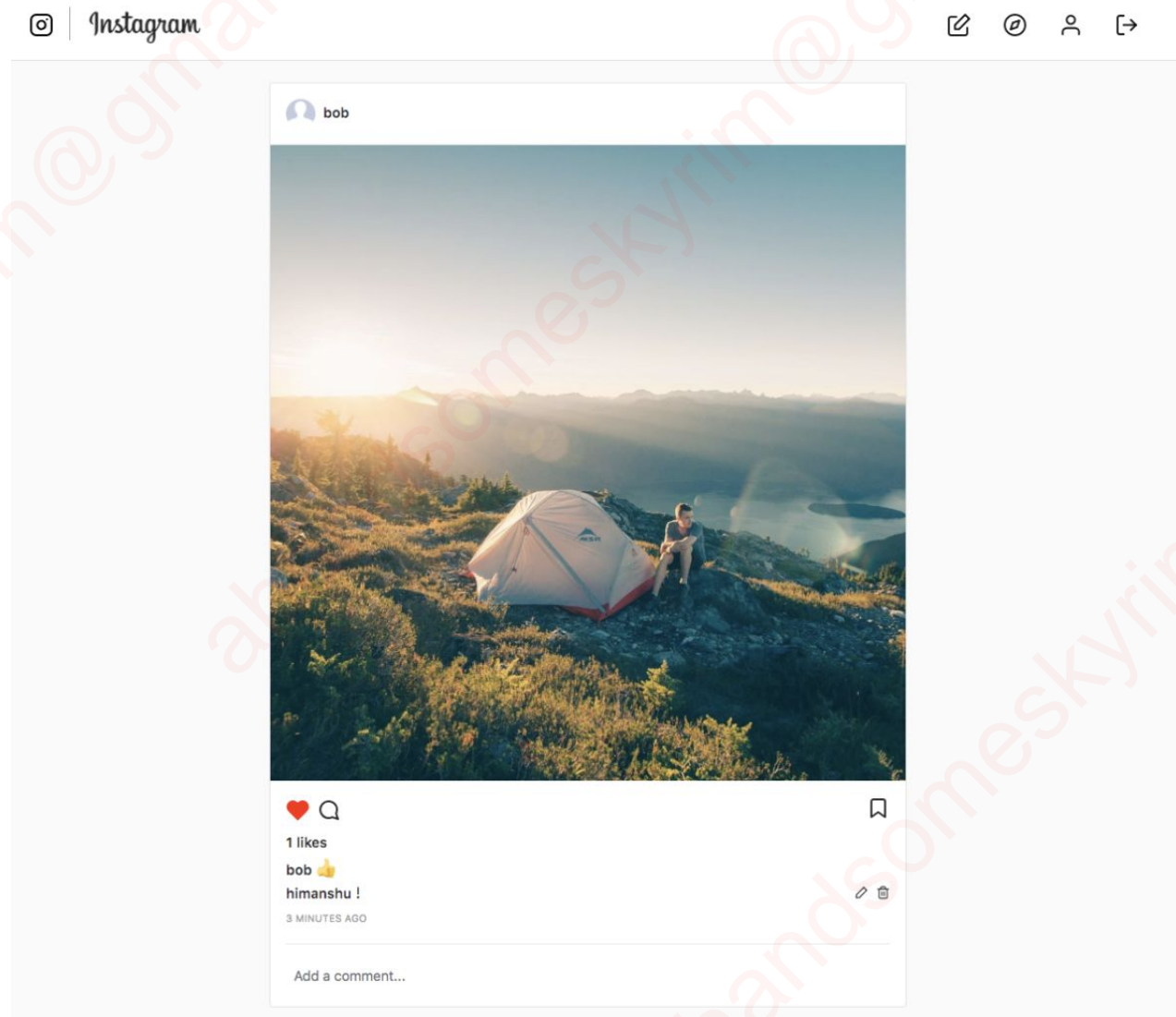
Do not have an account? [Sign up](#)

Instagram

Sign Up

Already have an account? [Log in](#)

2. 项目类型以及难度



3. 课程要求和作业

课程要求和作业

根据课程大纲提前预习,
课后复习知识点, 阅读相关技术文章
(Medium, dev.to, juejin.cn, zhihu.com)

coding

coding

coding!

HTML, CSS, JavaScript

按照要求完成作业, 一定要动手coding, 在tutorial上面tutor会解答问题

4. Web基础知识介绍

Internet的基本概念和作用

从构成的角度来看

节点:主机节点和数据交换节点

边:包括接入的链路和骨干链路。

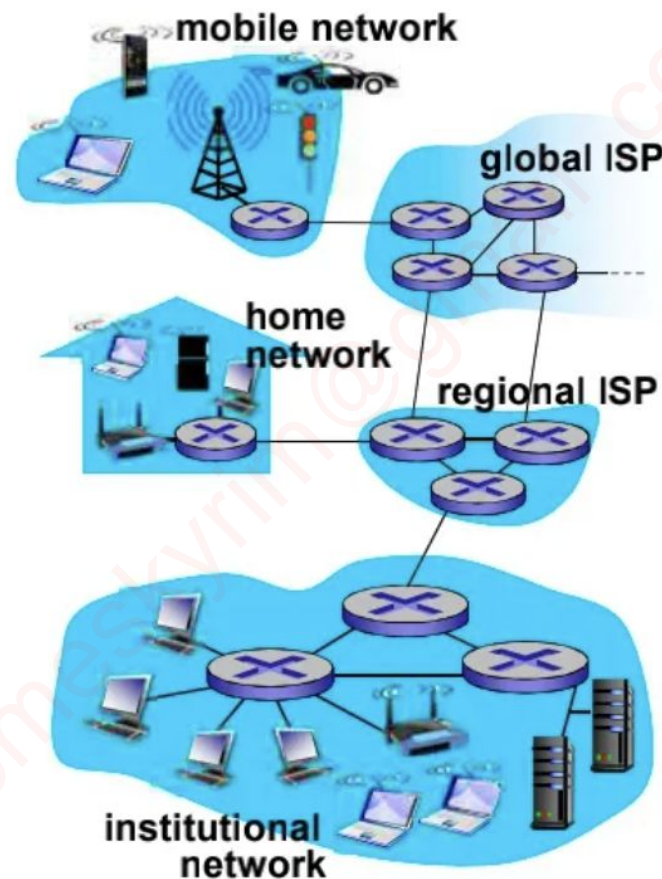
把一堆比较关联紧密的节点和边连在一起, 会构成一个个的网络

协议:人类协议 vs 网络协议

网络的网络:互联网是由一堆网络通过网络, 互联设备(路由器), 连接在一起的网络的网络。

从服务角度来看互联网

互联网就是分布式的应用, 以及为分布式应用提供通信服务的基础设施。



4. Web基础知识介绍

Internet的基本概念和作用

从构成的角度来看:

节点:主机节点和数据交换节点

边:包括接入的链路和骨干链路。

把一堆比较关联紧密的节点和边连在一起, 会构成一个个的网络

协议:人类协议 vs 网络协议

网络的网络:互联网是由一堆网络通过网络, 互联设备(路由器), 连接在一起的网络的网络。

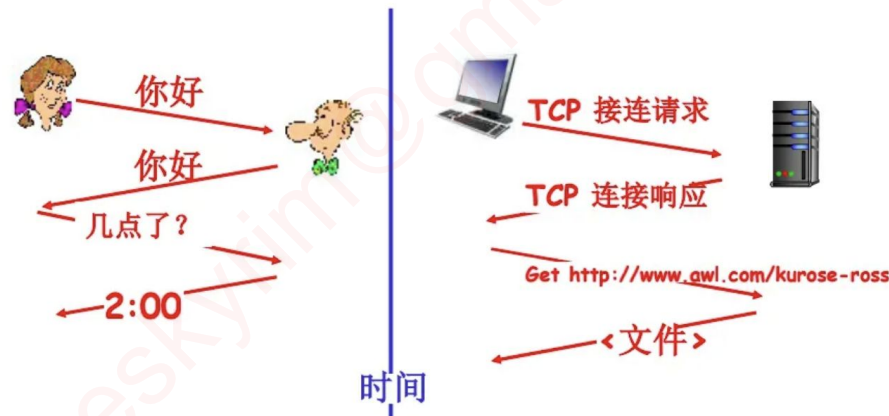
从服务角度来看互联网:

互联网就是分布式的应用, 以及为分布式应用提供通信服务的基础设施。

扩展:三次握手和四次挥手

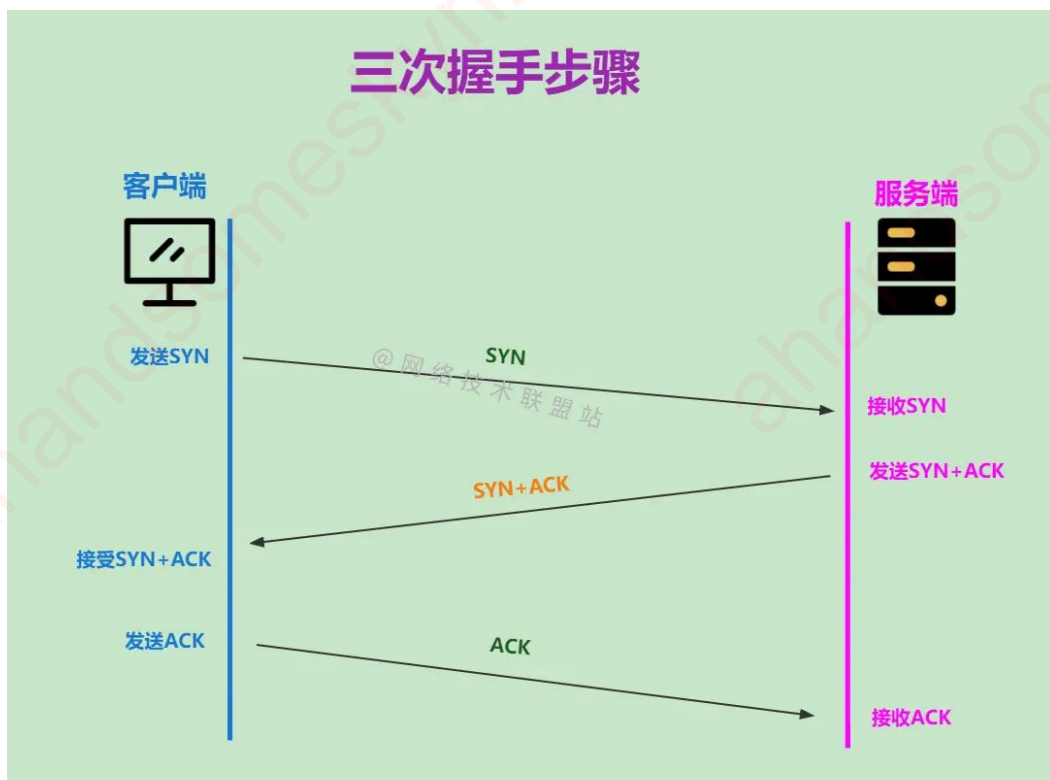
什么是协议?

人类协议和计算机网络协议示例



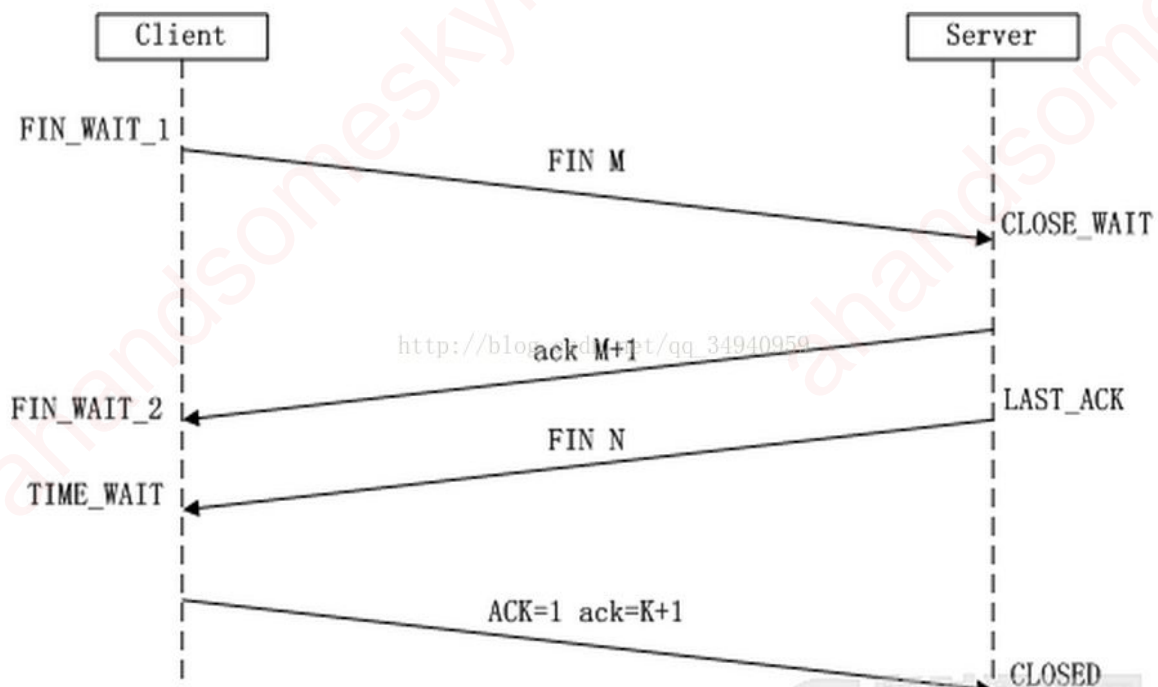
4. Web基础知识介绍

扩展: TCP三次握手和四次挥手



4. Web基础知识介绍

扩展: TCP三次握手和四次挥手



四次挥手——生活中例子：男女朋友分手



小美



小明

小明，相处那么长时间了，我觉得我们不合适，分手吧

别，我还有话和你说，“xxxxxx”

既然你想好了，我再纠缠你也没意思，那就和平分手吧~

那就这样了，互删微信吧，再见！

最多等待四分钟

4. Web基础知识介绍

TCP VS UDP

- TCP 是面向连接的、可靠的、有序的、速度慢的协议;UDP 是无连接的、不可靠的、无序的、速度快的协议。
- TCP 开销比 UDP 大, TCP 头部需要 20 字节, UDP 头部只要 8 个字节。
- TCP 无界有拥塞控制, UDP 有界无拥塞控制。
- 使用情况不同

	TCP	UDP
连接性	面向连接	无连接
可靠性	可靠	不可靠
有序性	有序	无序
有界性	有界	无界
拥塞控制	有	无
传输速度	慢	快
量级	重量级	轻量级
头部大小	大	小

4. Web基础知识介绍

统一资源定位符：URL

URL 是 统一资源定位符 (uniform resource locator) 的缩写，顾名思义就是用来定位网络资源的。

Web 应用中的 HTML 网页、JavaScript 源码、CSS 样式表以及图片等多媒体资源，都可以通过一个 URL 来访问。URL 唯一标识一个网络资源，并指明了获取它的方式。

4. Web基础知识介绍

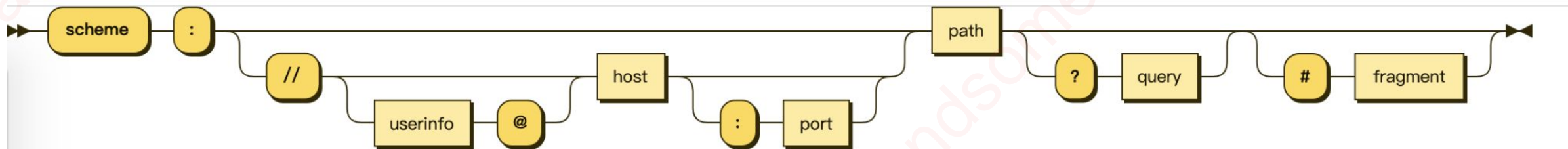
统一资源定位符: URL

<https://mui.com/material-ui/getting-started/#introduction>

https://www.google.com/search?q=jiangren&rlz=1C5GCEM_enAU1041AU1041&oq=jiangren&aqs=chrome.0.69i59j0i512j46i175i199i512j69i60l5.2302j0j15&sourceid=chrome&ie=UTF-8

- 模式 (*schema*), 相当于 *URL* 的类型, 通常用来表示访问协议, 比如 HTTP ;
- 用户信息 (*userinfo*), 用于访问资源的用户信息, 可选;
- 主机 (*host*), 资源所在主机, 可选;
- 端口 (*port*), 资源所在主机端口, 可选, 不填则使用协议的默认端口;
- 路径 (*path*), 资源所在路径;
- 查询 (*query*), 用于访问资源的参数, 比如网页传参, 可选;
- 片段 (*fragment*), 指定资源中的某个片段, 比如网页中的锚, 可选;

<https://learn.jiangren.com.au:443/my-courses?search=web&type=fulltime#detail>



4. Web基础知识介绍

URL常见形式

模式/协议	URL	说明	端口
HTTP	http://	表示通过 <i>HTTP</i> 协议访问(的网页)	80
HTTPS	https://	表示通过 <i>HTTPS</i> 协议访问(的网页)	443
FTP	ftp://	表示通过 <i>FTP</i> 协议访问(的文本文件)	20, 21
Mailto	mailto:kevin@jr.com	系统将自动调起邮箱客户端发件窗口, 向指定邮箱发邮件	

4. Web基础知识介绍

URL常见形式

后端服务连接数据库，有赖于数据库的服务器地址、端口、用户、密码等信息。这些信息可以组织成一个 URL，这就是所谓的数据库连接串：

[数据库类型]://[用户]:[密码]@[服务器地址]:[端口]/[库名]?[参数①]=[值①]&[参数②]=[值②]

下面是一个典型的数据库连接串，用来连接 MongoDB：

mongodb://test:123456@10.0.0.1:27017/demo?replicaSet=demo

- 数据库类型：MongoDB
- 服务器地址：10.0.0.1
- 端口：27017
- 认证用户：test
- 密码：123456
- 访问数据库名：demo

4. Web基础知识介绍

URL常见形式

后端服务连接数据库，有赖于数据库的服务器地址、端口、用户、密码等信息。这些信息可以组织成一个 URL，这就是所谓的数据库连接串：

[数据库类型]://[用户]:[密码]@[服务器地址]:[端口]/[库名]?[参数①]=[值①]&[参数②]=[值②]

下面是一个典型的数据库连接串，用来连接 MongoDB：

mongodb://test:123456@10.0.0.1:27017/demo?replicaSet=demo

- 数据库类型：MongoDB
- 服务器地址：10.0.0.1
- 端口：27017
- 认证用户：test
- 密码：123456
- 访问数据库名：demo

4. Web基础知识介绍

HTTP Methods <https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods>

方法	描述
GET	请求指定的页面信息，并返回 实体主体。
HEAD	类似于 GET 请求，只不过返回的响应中没有具体的内容，用于 获取报头
POST	向指定资源提交数据进行处理请求（例如提交表 单或者上传文件）。数据被包含在请求体中。POST 请求可能会导致新的资源的建立和/或已有资源的修改。
PUT	从客户端向服务器传送的数据取代指定的文档的内容。
DELETE	请求服务器删除指定的页面。
CONNECT	HTTP/1.1 协议中预留给能够将连接改为管道方式的代理服 务器。
OPTIONS	描述目标资源的通信选项。
TRACE	回显服务器收到的请求，主要用于测试或诊断。
PATCH	是对 PUT 方法的补充，用来对已知资源进行局部更新 。

4. Web基础知识介绍

HTTP Status Code <https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>

HTTP response status codes indicate whether a specific [HTTP](#) request has been successfully completed. Responses are grouped in five classes:

1. [Informational responses](#) (100 – 199)
2. [Successful responses](#) (200 – 299)
3. [Redirection messages](#) (300 – 399)
4. [Client error responses](#) (400 – 499)
5. [Server error responses](#) (500 – 599)

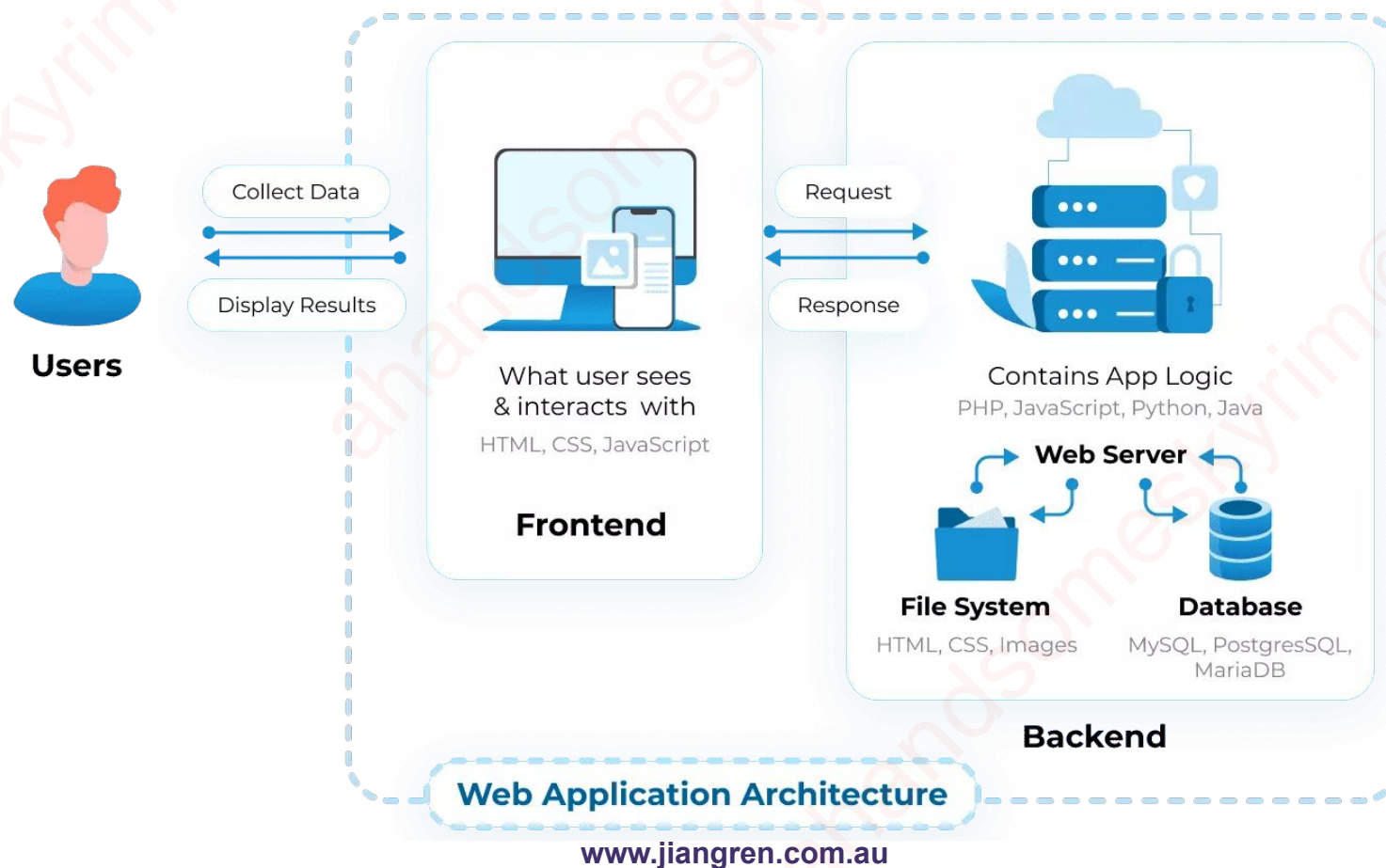
4. Web基础知识介绍

浏览器输入 URL 地址后发生了什么？

- URL 解析:浏览器首先解析输入的 **URL** 地址, 将其分解成几个组成部分, 包括协议、域名、路径、查询参数等。
- DNS 解析:浏览器将解析到的域名(或 IP地址)发送给**DNS**(域名系统)服务器, 以获取与域名相对应的IP地址。
- 建立连接:浏览器使用获取到的**IP**地址与目标服务器建立 **TCP** 连接。这是通过**三次握手**来确保双方建立可靠的连接。
- 发起请求:浏览器向目标服务器发送 **HTTP** 请求, 其中包含请求方法(如**GET**、**POST**)、路径、请求头和请求体等信息。
- 服务器响应:目标服务器接收到请求后, 处理请求并生成相应的**HTTP响应**。响应包括**响应状态码**、**响应头**和**响应体**等信息。
- 响应处理:浏览器接收到服务器的响应后, 根据响应头中的信息判断响应的内容类型, 并将响应体中的数据进行解析和处理。
- 渲染页面:如果响应的内容是HTML, 浏览器会解析**HTML**, 并根据其中的**CSS**和**JavaScript**等资源进行页面的渲染和呈现。
- 下载资源:在解析HTML时, 如果发现页面中引用了其他资源(如CSS、JavaScript、图像等), 浏览器会继续发送请求并下载这些资源。
连接结束时, 通过**四次挥手**结束TCP连接。
- 执行脚本:如果页面中包含**JavaScript**代码, 浏览器会执行这些脚本, 以实现交互、动态效果和数据处理等功能。
- 页面展示:最后, 浏览器将解析后的页面内容进行布局、绘制, 最终展示给用户, 用户可以在浏览器中浏览和与页面进行交互。

4. Web基础知识介绍

Web应用架构

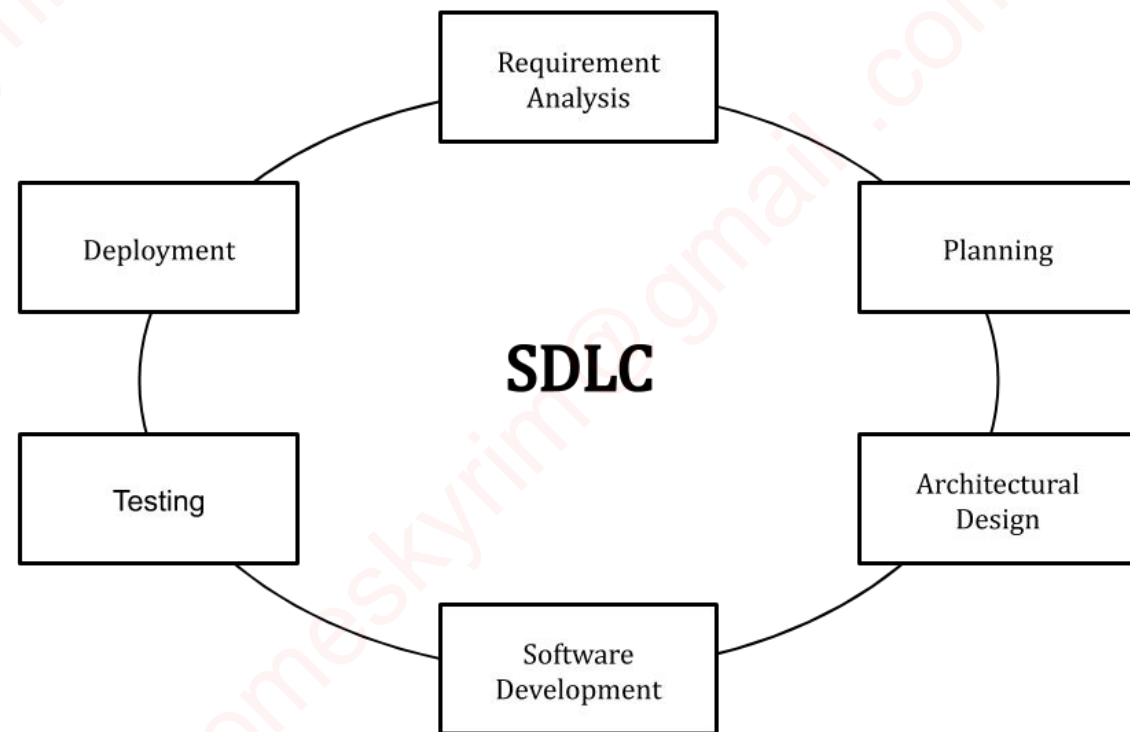


5. 开发流程介绍

软件开发过程

Software Development Life Cycle (SDLC)

- Requirement analysis
- Planning
- Software design such as architectural design
- Software development
- Testing
- Deployment



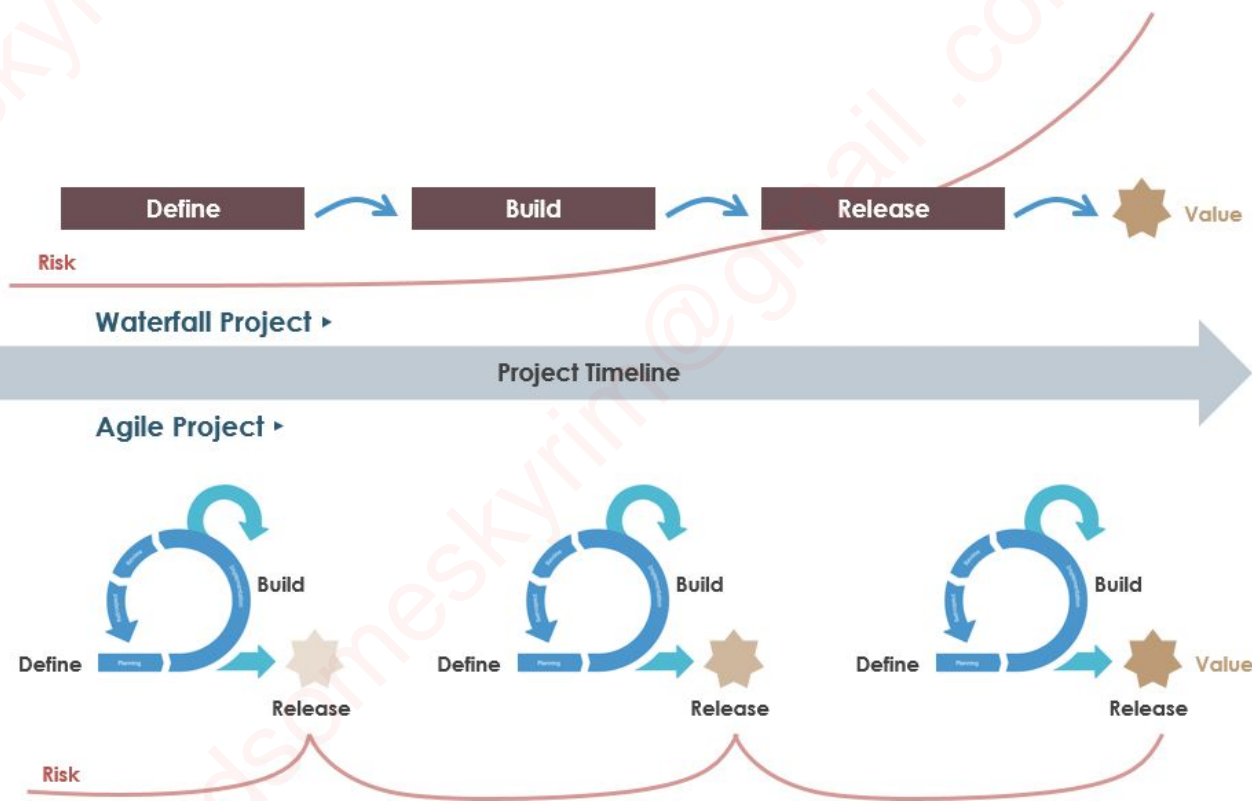
5. 开发流程介绍

软件开发过程

Software Development Life Cycle (SDLC)

瀑布模型 (Waterfall Model): 瀑布模型是一种线性顺序的开发过程, 按照阶段顺序执行, 每个阶段的结果作为下一阶段的输入。包括需求分析、系统设计、编码、测试和维护等阶段。

敏捷模型 (Agile Model): 敏捷开发强调迭代和增量开发, 将开发过程划分为多个短周期(2-4 weeks)的迭代, 每个迭代都交付可工作的软件。敏捷开发注重团队协作、持续交付和快速响应变化。





Homework

1. 复习URL的组成结构
2. 复习输入URL之后发生了什么
3. 注册medium, dev.to, juejin, zhihu
4. 下载vscode, mongodb server, nvm



Q & A