

前情概要

- 第一章
 - 重构极客时间网站，使用 Node.js 搭建 BFF 层。

前情概要

- 第二章
 - Node.js 基础
 - 向前提供 HTTP 服务
 - 向后进行 RPC 通信

极客时间整站与第一个页面的需求解构

整站与第一个页面的需求分析

- 路由
 - 每一个页面可以以一个独立的 Koa 应用开发
 - 实际项目开发中可以实现并行开发

整站与第一个页面的需求分析

- 静态页面渲染

整站与第一个页面的需求分析

- 整站规划架构图

极客时间课程 App 下载页的需求实现

极客时间课程详情页的需求解构

极客时间课程详情页的需求解构

- 使用 RPC 通信从后台服务器获取数据

极客时间课程详情页的需求解构

- 模板渲染
 - include 子模板
 - xss 过滤、模板 helper 函数

使用 ES6 模板字符串实现模板引擎

使用 ES6 模板字符串实现模板引擎

- 通过 vm 模块编译 JavaScript 形成函数
 - xss 过滤、模板 helper 函数
 - include 子模板

极客时间课程详情页的需求实现

极客时间课程播放页的需求解构

极客时间课程播放页的需求解构

- API 服务 - RESTful
 - 简单易懂
 - 可以快速搭建
 - 在数据的聚合方面有很大劣势

极客时间课程播放页的需求解构

- API 服务 - GraphQL
 - 专注数据聚合，前端要什么就返回什么

GraphQL

GraphQL

- Facebook 开发的实现 API 服务的库
 - 让前端有“自定义查询”数据的能力

GraphQL

- 查询语法
- 类型系统
- 校验系统

极客时间课程播放页的需求实现

极客时间课程列表页的需求解构

极客时间课程列表页的需求解构

- 后端需要渲染列表
 - 首屏加速
 - SEO
- 前端也需要渲染列表
 - 无刷新过滤、排序

极客时间课程列表页的需求解构

- 前后端同构
 - 同一个模板/组件，可在浏览器渲染也可在 Node.js 渲染。

使用 React/Vue 进行服务端渲染

使用 React/Vue 进行服务端渲染

- `ReactDOMServer.renderToString()`
- `VueServerRenderer.renderToString()`

使用 React/Vue 进行服务端渲染

- Next/Nuxt
 - 对 React/Vue 的前后端同构有一个完整的解决方案
 - 不太方便与现有服务组合

极客时间课程列表页的需求实现

整站回顾/整合