



(12) 发明专利申请

(10) 申请公布号 CN 118519619 A

(43) 申请公布日 2024. 08. 20

(21) 申请号 202410758671.8

(22) 申请日 2024.06.12

(71) 申请人 太保科技有限公司

地址 200010 上海市黄浦区中山南路1号11层(名义楼层)11V6室

(72) 发明人 邓秋爽

(74) 专利代理机构 北京集佳知识产权代理有限公司 11227

专利代理师 杨平平

(51) Int. Cl.

G06F 8/20 (2018.01)

G06F 8/73 (2018.01)

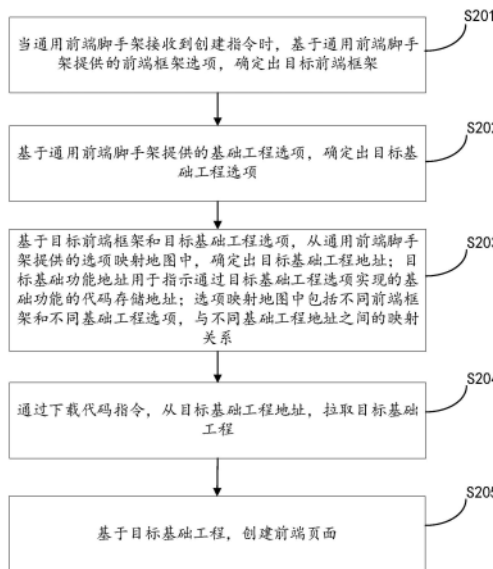
权利要求书2页 说明书11页 附图3页

(54) 发明名称

一种前端脚手架的应用方法及装置

(57) 摘要

本申请公开了一种前端脚手架的应用方法及装置,该方法应用于通用前端脚手架,该方法包括:当通用前端脚手架接收到创建指令时,基于通用前端脚手架提供的前端框架选项,确定出目标前端框架;基于通用前端脚手架提供的基础工程选项,确定出目标基础工程选项;基于目标前端框架和目标基础工程选项,从通用前端脚手架提供的选项映射地图中,确定出目标基础工程地址;目标基础功能地址用于指示通过目标基础工程选项实现的基础功能的代码存储地址;通过下载代码指令,从目标基础工程地址,拉取目标基础工程;基于目标基础工程,创建前端页面。该方法能够提供一种适配于多种前端框架的前端脚手架,从而提升开发效率。



1. 一种前端脚手架的应用方法,其特征在于,应用于通用前端脚手架,所述方法包括:
当所述通用前端脚手架接收到创建指令时,基于所述通用前端脚手架提供的前端框架选项,确定出目标前端框架;
基于所述通用前端脚手架提供的基础工程选项,确定出目标基础工程选项;
基于所述目标前端框架和所述目标基础工程选项,从所述通用前端脚手架提供的选项映射地图中,确定出目标基础工程地址;所述目标基础功能地址用于指示通过所述目标基础工程选项实现的基础功能的代码存储地址;所述选项映射地图中包括不同前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系;
通过下载代码指令,从所述目标基础工程地址,拉取目标基础工程;
基于所述目标基础工程,创建前端页面。
2. 根据权利要求1所述的方法,其特征在于,所述基础工程选项包括构建工具、用户界面组件库、路由封装选项、请求服务封装选项以及数据管理库封装选项。
3. 根据权利要求1所述的方法,其特征在于,所述前端页面包括快速开发页面,所述基于所述目标基础工程,创建前端页面,包括:
基于所述目标基础工程,通过创建页面命令中的预设参数,创建快速开发页面,所述快速开发页面是由所述通用前端脚手架提供的页面模式。
4. 根据权利要求3所述的方法,其特征在于,所述创建快速开发页面包括:
基于所述目标前端框架,从所述通用前端脚手架提供的页面模板映射地图中,确定出所述快速开发页面对应的快速开发页面模板;所述页面模板映射地图中包括不同前端框架与不同页面模板之间的映射关系;所述快速开发页面模板中包括表格组件和表单组件,以及表格组件和表单组件分别对应的基础交互操作;
基于所述快速开发页面模板和结构化数据配置文件,得到所述快速开发页面。
5. 根据权利要求4所述的方法,其特征在于,所述创建页面命令中包括所述快速开发页面的路由地址,所述方法还包括:
通过基于所述快速开发页面模板和所述快速开发页面的路由地址,生成对应的代码文件,将所述代码文件存储于所述目标基础工程对应的文件夹中。
6. 根据权利要求5所述的方法,其特征在于,所述方法还包括:
通过获取命令从所述目标基础工程对应的文件夹中读取存储文件,基于所述目标基础工程为所述存储文件划分存储位置。
7. 根据权利要求1所述的方法,其特征在于,所述基于所述目标基础工程,创建前端页面之前,所述方法还包括:
通过命令行指令启动子进程,所述子进程用于安装所述目标基础工程的相关依赖和启动服务器。
8. 一种前端脚手架的应用装置,其特征在于,部署于通用前端脚手架,所述装置包括:
第一确定模块,用于当所述通用前端脚手架接收到创建指令时,基于所述通用前端脚手架提供的前端框架选项,确定出目标前端框架;
第二确定模块,用于基于所述通用前端脚手架提供的基础工程选项,确定出目标基础工程选项;
第三确定模块,用于基于所述目标前端框架和所述目标基础工程选项,从所述通用前

端脚手架提供的选项映射地图中,确定出目标基础工程地址;所述目标基础功能地址用于指示通过所述目标基础工程选项实现的基础功能的代码存储地址;所述选项映射地图中包括不同前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系;

拉取模块,用于通过下载代码指令,从所述目标基础工程地址,拉取目标基础工程;

创建模块,用于基于所述目标基础工程,创建前端页面。

9.一种前端脚手架的应用设备,其特征在于,所述设备包括:

存储器,用于存储计算机程序;

处理器,用于执行所述计算机程序,以使所述设备执行如权利要求1至7任一项所述前端脚手架的应用方法。

10.一种计算机存储介质,其特征在于,所述计算机存储介质中存储有计算机程序,所述计算机程序被处理器执行时实现如权利要求1至7任一项所述前端脚手架的应用方法。

一种前端脚手架的应用方法及装置

技术领域

[0001] 本申请涉及数据开发技术领域,尤其涉及一种前端脚手架的应用方法及装置。

背景技术

[0002] 前端脚手架是一个工具集,用于快速搭建和开发前端项目,旨在提高前端开发效率和规范化项目结构。前端脚手架通常包含项目初始化、构建、打包、部署等一系列工程化的流程和工具,帮助开发者快速搭建某种前端框架的项目的基础结构。

[0003] 随着前端工程化的发展,在前端开发领域中,前端脚手架逐渐走进人们的视野中,前端脚手架可以帮助开发者快速搭建项目结构、自动化工作流程、提供配置项等,从而提高了开发效率,且保证了前端项目的规范性和一致性。

[0004] 目前,脚手架工具都由前端框架开发者提供,如果在前端开发过程中,选择不同的前端开发框架进行开发时,则需要安装对应的脚手架工具,并且各个脚手架工具的使用方法均不同,缺乏一种适用于大部分前端框架的前端脚手架,导致基于不同前端框架进行开发时的开发效率差。

发明内容

[0005] 有鉴于此,本申请实施例提供了一种前端脚手架的应用方法及装置,旨在提供一种适配于多种前端框架的前端脚手架,从而提升开发效率。

[0006] 第一方面,本申请实施例提供了一种前端脚手架的应用方法,应用于通用前端脚手架,所述方法包括:

[0007] 当所述通用前端脚手架接收到创建指令时,基于所述通用前端脚手架提供的前端框架选项,确定出目标前端框架;

[0008] 基于所述通用前端脚手架提供的基础工程选项,确定出目标基础工程选项;

[0009] 基于所述目标前端框架和所述目标基础工程选项,从所述通用前端脚手架提供的选项映射地图中,确定出目标基础工程地址;所述目标基础功能地址用于指示通过所述目标基础工程选项实现的基础功能的代码存储地址;所述选项映射地图中包括不同前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系;

[0010] 通过下载代码指令,从所述目标基础工程地址,拉取目标基础工程;

[0011] 基于所述目标基础工程,创建前端页面。

[0012] 可选地,所述基础工程选项包括构建工具、用户界面组件库、路由封装选项、请求服务封装选项以及数据管理库封装选项。

[0013] 可选地,所述前端页面包括快速开发页面,所述基于所述目标基础工程,创建前端页面,包括:

[0014] 基于所述目标基础工程,通过创建页面命令中的预设参数,创建快速开发页面,所述快速开发页面是由所述通用前端脚手架提供的页面模式。

[0015] 可选地,所述创建快速开发页面包括:

[0016] 基于所述目标前端框架,从所述通用前端脚手架提供的页面模板映射地图中,确定出所述快速开发页面对应的快速开发页面模板;所述页面模板映射地图中包括不同前端框架与不同页面模板之间的映射关系;所述快速开发页面模板中包括表格组件和表单组件,以及表格组件和表单组件分别对应的基础交互操作;

[0017] 基于所述快速开发页面模板和结构化数据配置文件,得到所述快速开发页面。

[0018] 可选地,所述创建页面命令中包括所述快速开发页面的路由地址,所述方法还包括:

[0019] 通过基于所述快速开发页面模板和所述快速开发页面的路由地址,生成对应的代码文件,将所述代码文件存储于所述目标基础工程对应的文件夹中。

[0020] 可选地,所述方法还包括:

[0021] 通过获取命令从所述目标基础工程对应的文件夹中读取存储文件,基于所述目标基础工程为所述存储文件划分存储位置。

[0022] 可选地,所述基于所述目标基础工程,创建前端页面之前,所述方法还包括:

[0023] 通过命令行指令启动子进程,所述子进程用于安装所述目标基础工程的相关依赖和启动服务器。

[0024] 第二方面,本申请实施例提供了一种前端脚手架的应用装置,部署于通用前端脚手架,所述装置包括:

[0025] 第一确定模块,用于当所述通用前端脚手架接收到创建指令时,基于所述通用前端脚手架提供的前端框架选项,确定出目标前端框架;

[0026] 第二确定模块,用于基于所述通用前端脚手架提供的基础工程选项,确定出目标基础工程选项;

[0027] 第三确定模块,用于基于所述目标前端框架和所述目标基础工程选项,从所述通用前端脚手架提供的选项映射地图中,确定出目标基础工程地址;所述目标基础功能地址用于指示通过所述目标基础工程选项实现的基础功能的代码存储地址;所述选项映射地图中包括不同前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系;

[0028] 拉取模块,用于通过下载代码指令,从所述目标基础工程地址,拉取目标基础工程;

[0029] 创建模块,用于基于所述目标基础工程,创建前端页面。

[0030] 第三方面,本申请实施例提供了一种前端脚手架的应用设备,所述设备包括:

[0031] 存储器,用于存储计算机程序;

[0032] 处理器,用于执行所述计算机程序,以使所述设备执行前述第一方面所述的前端脚手架的应用方法。

[0033] 第四方面,本申请实施例提供了一种计算机存储介质,所述计算机可读存储介质上存储有计算机程序,当所述计算机程序被运行时,运行所述计算机程序的设备实现前述第一方面所述的前端脚手架的应用方法。

[0034] 相较于现有技术,本申请实施例具有以下有益效果:

[0035] 本申请实施例提供了一种前端脚手架的应用方法及装置,该方法应用于通用前端脚手架,该方法可以包括:当通用前端脚手架接收到创建指令时,可以基于通用前端脚手架提供的前端框架选项,从中选择出目标前端框架。然后,可以基于通用前端脚手架提供的基

础工程选项,从中确定出目标基础工程选项。选项映射地图中包括不同前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系。因此,可以在确定出目标前端框架和目标基础工程选项后,可以根据目标前端框架和目标基础工程选项,从通用前端脚手架提供的选项映射地图中,确定出用于通过目标基础工程选项实现的基础工程的代码存储地址,即目标基础工程地址。之后,可以通过下载代码指令,从映射出的目标基础工程地址中,拉取到目标基础工程,从而可以基于目标基础工程,创建前端页面。由此,通过通用前端脚手架完成了项目初始化,可以基于通用前端脚手架提供的目标基础工程,继续进行前端开发。

[0036] 可见,通用前端脚手架中提供了选项映射地图,其中包含有各个前端框架和不同基础工程选项,与目标基础工程地址的映射关系,使得通用前端脚手架提供的基础工程可以适用于多种前端框架,并且能够通过目标基础工程选项实现对应的基础工程,从而基于目标基础工程可以继续前端开发,无需安装多种前端框架对应的脚手架,进而提升了开发效率。

附图说明

[0037] 为更清楚地说明本实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本申请的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0038] 图1为本申请实施例提供的一种前端脚手架的应用方法的应用场景;

[0039] 图2为本申请实施例提供的一种前端脚手架的应用方法的流程图;

[0040] 图3为本申请实施例提供的一种前端脚手架的应用方法的示意图;

[0041] 图4为本申请实施例提供的一种前端脚手架的应用装置的结构示意图。

具体实施方式

[0042] 为了使本技术领域的人员更好地理解本申请方案,下面将结合本申请实施例中的附图,对本申请实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅是本申请一部分实施例,而不是全部的实施例。基于本申请中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本申请保护的范围。

[0043] 目前,脚手架工具可以实现项目初始化、插件管理和项目配置,项目初始化是指可以通过脚手架工具快速创建一个对应框架的项目的基础结构,包括目录结构、配置文件等。插件管理是指脚手架工具支持插件系统,开发者可以通过插件扩展项目功能,例如添加路由、状态管理等。项目配置是指允许开发者通过脚手架工具对项目进行自定义配置,包括构建工具配置、编译器Babel配置等,以满足项目的特定需求。

[0044] 目前,脚手架工具都由前端框架开发者提供,如果在前端开发过程中,选择不同的前端开发框架进行开发时,则需要安装对应的脚手架工具,并且各个脚手架工具的使用方法均不同,缺乏一种适用于大部分前端框架的前端脚手架,导致基于不同前端框架进行开发时的开发效率差。

[0045] 基于此,为了解决上述问题,在本申请实施例中,当通用前端脚手架接收到创建指令时,可以基于通用前端脚手架提供的前端框架选项,从中选择出目标前端框架。然后,可

以基于通用前端脚手架提供的基础工程选项,从中确定出目标基础工程选项。选项映射地图中包括不同前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系。因此,可以在确定出目标前端框架和目标基础工程选项后,可以根据目标前端框架和目标基础工程选项,从通用前端脚手架提供的选项映射地图中,确定出用于通过目标基础工程选项实现的基础工程的代码存储地址,即目标基础工程地址。之后,可以通过下载代码指令,从映射出的目标基础工程地址中,拉取到目标基础工程,从而可以基于目标基础工程,创建前端页面。由此,通过通用前端脚手架完成了项目初始化,可以基于通用前端脚手架提供的目标基础工程,继续进行前端开发。通用前端脚手架中提供了选项映射地图,其中包含有各个前端框架和不同基础工程选项,与目标基础工程地址的映射关系,使得通用前端脚手架提供的基础工程可以适用于多种前端框架,并且能够通过目标基础工程选项实现对应的基础工程,从而基于目标基础工程可以继续前端开发,无需安装多种前端框架对应的脚手架,进而提升了开发效率。

[0046] 举例来说,本申请实施例的场景之一,可以是应用到如图1所示的场景中。该场景包括终端101,其中,终端101中运行有通用前端脚手架,以基于通用前端脚手架提供的选项映射地图中,确定出目标基础工程地址,以拉取到目标基础工程,进而可以基于目标基础工程,创建前端页面。

[0047] 首先,在上述应用场景中,虽然将本申请实施例提供的实施方式的动作描述由终端101执行;但是,本申请实施例在执行主体方面不受限制,只要执行了本申请实施例提供的实施方式所公开的动作即可。

[0048] 其次,上述场景仅是本申请实施例提供的一个场景示例,本申请实施例并不限于此场景。

[0049] 下面结合附图,通过实施例来详细说明本申请实施例中前端脚手架的应用方法及装置的具体实现方式。

[0050] 参见图2,该图为本申请实施例提供的一种前端脚手架的应用方法的流程图,该方法应用于通用前端脚手架,结合图2所示,具体可以包括:

[0051] S201:当通用前端脚手架接收到创建指令时,基于通用前端脚手架提供的前端框架选项,确定出目标前端框架。

[0052] 通用前端脚手架是本申请实施例提供的一种脚手架工具,通过该通用前端脚手架可以用于项目初始化。具体地,该通用前端脚手架中提供了构建工具、用户界面组件(User Interface,UI)库、路由封装选项、请求服务封装选项以及数据管理库封装选项,以供用户根据通用前端脚手架提供的上述基础工程选项进行选择。其中,构建工具中包括基础构建工具,例如,打包工具webpack、模块打包器Rollup以及流程自动化工具Grunt等。UI库中包括基础组件,例如,按钮、导航栏、弹出框、标签页以及下拉菜单等基础组件。路由封装选项中包括通用前端脚手架提供的默认两级路由,并且提供路由鉴权接口,可以由开发者自定义路由鉴权方法,并且在需要三级路由的情况下,可以由开发者自行扩展。其中,路由鉴权接口是指在前端应用程序中通过特定的接口实现对路由访问权限的鉴定和控制,用于确定用户是否具有权限访问某个页面或路由。

[0053] 请求服务封装选项是指通用前端脚手架提供了基于传统网络请求技术(XMLHttpRequest,XHR)和基于网路请求应用程序接口Fetch的多种请求库以供开发者进行

选择。通过封装请求服务,可以隐藏底层实现细节,简化接口调用,并提供更高层次的抽象,有助于代码的可维护性和扩展性。然后,可以通用前端脚手架可以根据所选择的请求库自动生成请求服务封装代码,请求服务封装代码可以包括:基础请求封装、接口鉴权(与路由鉴权类似,鉴权方法可以由开发者自定义)、报错处理机制(封装有常见错误码的处理,可以修改该报错处理机制,还可以扩展该报错处理机制),以及文件上传或下载功能封装(可显示上传下载进度,支持大文件断点续传,即在文件传输过程中发生中断的情况下,能够记录已经传输的部分,并在连接恢复后从中断处继续传输)。

[0054] 数据管理库封装选项是指通用前端脚手架提供的数据库的选项,即用于管理项目中数据的库,便于开发者通过简便的方式处理数据的获取、存储、更新和展示等操作,便于开发者更好地组织和管理项目中的数据。然后,可以通过开发者选择的数据库自动生成数据管理封装代码,该状态管理库提供的基础功能包括:数据存储、数据获取、数据持久化。并且,数据管理库中提供了单一的数据流向,可以按照各个页面的命名进行数据隔离,避免数据流混乱造成问题定位困难。同时,基于上述基础工程选项,保证了通用前端脚手架的完整性。

[0055] 另外,通用前端脚手架还提供了快速开发页面模式,可以快速开发表格表单功能,通过一个指令即可根据用户选择的UI库,在封装有常规表格表单的搜索、分页、新增、编辑以及删除功能的快速开发页面模板中,自动生成对应的代码文件。对于具有相似功能的页面,无需重复开发,可以直接使用自动生成的代码文件,只需要根据实际业务需求,通过配置相应的结构化数据配置文件json文件即可得到功能完整的前端页面,完成了整个页面的开发。

[0056] 作为一种示例,在安装通用前端脚手架之后,首先可以通过通用前端脚手架进行项目初始化,即搭建初始项目工程。其中,通用前端脚手架可以发布于软件包管理工具npm,全局安装后便可应用于前端项目中。并且,在本申请实施例提供的通用前端脚手架中可以用于控制台选择的工具为命令行交互工具inquirer、可以用于处理控制台命令的工具为用于创建命令行界面的库commander,通用前端脚手架中的模板工具为嵌入式脚本语言JavaScript模板ejs。在通用前端脚手架中通过commander封装指令,例如,可以对创建指令create,创建页面指令create page,创建快速开发页面指令create page fast均使用commander接收用户指令。

[0057] 创建指令是指用于创建前端项目的指令,创建指令可以为create,通用前端脚手架接收到create指令后,可以通过inquirer提供控制台框架选项,即前端框架选项React、Angular、Vue2以及Vue3等不同种类的前端框架。由此,开发者可以从前端框架选项中选择一个前端框架,作为目标前端框架,以根据该目标前端框架确定出对应的基础工程,并基于该目标前端框架进行前端开发。

[0058] 另外,在通用前端脚手架中还支持开发者扩展前端框架,若当前前端框架中不包含开发者所需的前端框架时,可以有开发者自行扩展所需的前端框架。同时,通用前端脚手架中还支持由开发者自定义基础工程地址。其中,基础工程地址是指项目地址,可以由开发者自定义,也可以通过通用前端脚手架中获取。

[0059] S202:基于通用前端脚手架提供的基础工程选项,确定出目标基础工程选项。

[0060] 基础工程选项是指由通用前端脚手架提供的用于构建项目基础工程的选项,例

如,构建工具、UI库、路由封装选项、请求服务封装选项以及数据管理库封装选项等。目标基础工程选项是指由开发者从通用前端脚手架提供的基础工程选项中选择出的基础工程选项,例如,从请求服务封装选项中选择基于XHR的请求库作为目标基础工程选项中的请求库。

[0061] 例如,在确定出目标前端框架之后,通用前端脚手架可以根据目标前端框架提供对应的构建工具以及UI库选项等。并且,通用前端脚手架还提供有基于XHR和基于Fetch的多种请求库选项、层叠样式表(Cascading Style Sheets,css)编译器选项等。开发者可以根据个人开发习惯或者实际项目要求进行选择。

[0062] S203:基于目标前端框架和目标基础工程选项,从通用前端脚手架提供的选项映射地图中,确定出目标基础工程地址;目标基础功能地址用于指示通过目标基础工程选项实现的基础功能的代码存储地址;选项映射地图中包括不同前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系。

[0063] 目标基础工程地址是指从通用前端脚手架提供的选项映射地图中确定出的基础工程地址,用于指示通过目标基础工程选项实现的基础功能的代码存储地址。其中,通过目标基础工程选项实现的基础功能是指基于开发者选择的基础工程选项可以提供项目中的基础工具以构建出初始软件项目,例如,构建工具、请求封装以及编译器等。

[0064] 选项映射地图是由通用前端脚手架提供的,包括不同种前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系。例如,选项映射地图中包括当前端框架为Vue3框架时和构建工具为webpack时,基础工程地址为A;当前端框架为Vue2框架时和构建工具为vite时,基础工程地址为B。当目标前端框架为Vue3,且构建工具为webpack时,则可以从选项映射地图中确定出目标基础工程地址为A。

[0065] 因此,可以根据开发者选择的目标前端框架和目标基础工程选项,从选项映射地图中确定出基础工程地址,若开发者自定义基础工程地址,则可以省略该步骤,直接使用开发者提供的基础工程地址进行前端开发。

[0066] S204:通过下载代码指令,从目标基础工程地址,拉取目标基础工程。

[0067] 下载代码指令是指用于从基础工程地址中下载对应的基础工程代码的指令,例如,下载代码指令可以为download-git-repo。在通用前端脚手架确定出目标基础工程地址的情况下,可以通过上述下载代码指令拉取对应的目标基础工程。目标基础工程是指一个软件项目中的基础架构或基础设施部分,用于为软件项目提供基础设施,如路由封装、请求服务封装以及数据管理库封装等,以及提供UI库可以被整个项目共享和复用,还可以提供基础工具,如构建工具等。

[0068] 在拉取到目标基础工程之后,可以通过commander的创建子进程命令行child_process自动开启子进程,以安装目标基础工程的相关依赖并启动服务器,以在本地运行该项目。需要说明的是,在安装依赖和启动服务器的过程中可以在子进程中进行异步执行,避免阻塞,提升运行效率。

[0069] S205:基于目标基础工程,创建前端页面。

[0070] 最后,在获取到目标基础工程后,便完成了项目初始化,然后就可以基于目标基础工程,继续创建前端页面。

[0071] 另外,通用前端脚手架提供的快速开发页面模式封装于基础工程中,均放置于开

源平台,开发者可以根据项目需求进行扩展。

[0072] 可参照图3,图3为本申请实施例提供的一种前端脚手架的应用方法的示意图,在接收到创建指令之后,可以从通用前端脚手架中选择目标前端框架,也可以由开发者自定义框架,从而根据开发者自定义的框架创建对应的项目。

[0073] 在基于目标基础工程,创建前端页面的过程中,所创建的前端页面可以分为两种模式,第一种,可以基于目标基础工程创建基础页面,即仅提供目标前端框架的最基础页面,不提供封装与数据处理功能;第二种,可以基于目标基础工程创建快速开发页面。在创建页面时,可以通过创建页面命令中的预设参数区分上述两种情况,以确定出目前新建的页面是基础页面还是快速开发页面。其中,创建页面命令可以为create page,预设参数可以为fast,当创建页面命令为create page fast时,则认为当前新建的页面为快速开发页面。

[0074] 在创建快速开发页面时,首先可以根据创建页面命令中的预设参数进行判断,然后可以为该快速开发页面创建路由地址。如create page fast<二级路由(菜单命名)><一级路由(父菜单)>[-d指定路径],其中,二级路由地址可以选择填写,不作为必填项。由于路由地址不可重复,开发者输入的一级路由地址和二级路由地址可以被通用前端脚手架作为页面的唯一标识,后续均可按照该标识进行模块划分等。

[0075] 需要说明的是,在创建快速开发页面时,若已经指定路径则在指定路径下生成文件夹,若没有指定路径,则默认放在视图views文件夹下。同时,可以在views/一级路由/二级路由/配置config,在配置文件中生成配置文件config,并且开发者没有额外开发需求时,在创建快速开发页面之后,配置config文件,即可完成页面创建,得到快速开发页面。同时,快速开发页面中还需要配置其他功能时,可以在生成的页面中由开发者自行扩展开发。

[0076] 作为一种示例,当通用前端脚手架接收到新建页面(创建页面命令)或新建组件命令后,以创建快速开发页面为例,具体的实施方式可以包括:

[0077] 首先,可以创建服务server文件,即创建一个用于处理服务器处理请求的服务器文件,并在这个服务文件中进行请求的封装操作。然后,可以基于目标前端框架,从通用前端脚手架提供的页面模板映射地图中,确定出快速开发页面对应的快速开发页面模板,即通用前端脚手架提供的ejs模板。其中,页面模板映射地图中包括不同前端框架与不同页面模板之间的映射关系。快速开发页面模板中包括表格组件和表单组件,以及表格组件和表单组件分别对应的基础交互操作,例如,增加、删除、修改、查询,以及分页等功能。例如,当前端框架为Vue2时,对应的页面模板为A模板;当前端框架为Vue3时,对应的页面模板为B模板。

[0078] 需要说明是,由开发者自行扩展的前端框架,此时可以由开发者自行扩展对应的页面模板。

[0079] 由于创建页面命令中包括快速开发页面的路由地址,并且该路由地址可以作为快速开发页面的唯一标识,用于模块划分,在基于快速开发页面模板生成代码的过程中,可以基于快速开发页面模板和快速开发页面的路由地址生成对应的代码文件,即将快速开发页面模板中的占位符替换为该唯一标识,然后通过开发服务端应用程序node文件处理模块生成对应的文件,若接收到指定路径,则将该文件存储于指定路径,否则,将该文件存储于默认文件夹下,即将代码文件存储于目标基础工程对应的文件夹中。

[0080] 然后,在基于快速开发模板生成代码的过程中,以目标基础工程中的前端框架为Vue且构建工具为webpack为例,可以创建数据流处理store,以存储和管理项目中的状态数据,统一管理数据的获取、修改和更新等。目标基础工程可以通过获取命令require从目标基础工程对应的文件夹中读取存储文件,即读取本地store文件,从目标基础工程中引入开发者选择的数据管理库,统一管理存储文件中的数据,并且,可以按照生成的代码文件的路由地址为存储文件划分存储位置,即可以按照代码文件的唯一标识在存储文件中为生成的代码文件单独设置一个存储位置,实现了按照路由地址的分模块隔离。其他页面的模块划分可以由开发者根据实际项目需求进行确定,对此,本申请不具体限其他页面的存储文件中的存储位置划分。

[0081] 之后,可以目标基础工程可以通过require读取本地所有路由文件,以将这些路由文件映射到菜单上,从而在创建快速开发页面中实现创建路由。作为一种示例,菜单可以为一个网站的导航菜单或其他形式的菜单,便于开发者在网络或应用程序中浏览不同的页面或功能。在创建路由之后,可以通过写入方法fs.writeFile写入应用程序编程接口(Application Programming Interface,API)文件,即将定义API接口的数据以某种格式写入到文件中,以实现持久化存储并且便于后续读取处理。

[0082] 并且,在接口配置过程中,server服务引入了服务接口配置文件service/api.json文件中对应的接口对象,以生成快速开发页面中对应的对象,封装有对应的增删改查接口。例如,查询接口"query": "",删除接口"deleteItem": "",新增接口"create": "",以及修改接口"edit": ""。

[0083] 并且,server服务会在对应的store文件中引入,将请求服务数据写入对应的数据管理模块,即在存储文件中设置的存储位置。若此时开发者没有额外的需求,可以直接根据确定出的快速开发页面模板,并配置同时生成的结构化数据json配置文件,即可获得快速开发页面。其中,json配置文件只有一套,所有前端框架统一。可见,根据快速开发页面模板,以及相应地命令,可以自定生成完整页面或组件,只需完成后续json文件的配置,对于后台管理类包括多种表格表单的系统来说,提升了创建快捷性。

[0084] 另外,目标基础工程还提供了增删改查封装功能,例如,编辑弹框后数据回显、删除新增后刷新列表、搜索项改变后表格刷新等操作均已依赖于组件通信,例如,Vue3中可以使用钩子函数hooks进行调用,Vue2中可以使用事件总线eventbus实现。

[0085] 在完成上述接口配置之后,可以创建快速开发页面,可以在基于快速开发模板生成代码的过程中,通过目标基础工程封装可配置表格项的表格组件,并且能够封装可配置表单项的表单组件,使用对应的config文件配置生成表格组件和表单组件。然后,可以创建用于配置页面的json文件,例如,在表格配置过程中,自动生成以下配置,如:

```
[0086] contentTableConfig= {
[0087]   title: '用户列表',
[0088]   pageName: 'courseSource',
[0089]   propList: [
[0090]     {prop: 'name', label: '用户名', minWidth: '100'},
[0091]   ],
[0092]   showIndexColumn: true,
```

[0093] showSelectColumn:true

[0094] }

[0095] 上述代码结构介绍了用于配置内容表格的一些参数,例如,配置的标题、配置的页面名、配置列表的数组等信息。

[0096] 开发者在此配置过程中,只需要配置Label:表格列;prop:与后端字段对应的字段名,上述所提供的可配置功能,可以在目标基础工程的类型脚本ts文件中体现。

[0097] 例如,在表单配置过程中,自动生成以下配置,如:

[0098] searchFormConfig= {

[0099] pageName:"courseSource",

[0100] formItems:[{

[0101] field:"code",

[0102] type:"input",

[0103] label:"编码",

[0104] placeholder:"请输入编码",

[0105] maxlength:20

[0106]]],

[0107] itemTextAlign:'left',

[0108] labelWidth:"60px",

[0109] itemStyle:{padding:'10px 30px 10px 0'},

[0110] collayout:{span:8}

[0111] };

[0112] 上述代码结构介绍了用于配置内容表单的一些参数,例如,配置的页面名称、包含表单项配置的数组、表单项文本内容的对齐方式、表单项标签的宽度,表单的样式设置等信息。与上述表格配置相同,上述所提供的可配置功能,可以在目标基础工程的类型脚本ts文件中体现。

[0113] 需要说明的是,除了上述目标基础工程提供功能之外,对于由用户自定义的表格表单项,对应的页面文件中包括自定义插槽,对应地,自动生成文档中包括相关注释以提示插槽位置,用于开发者自定义表格表单项。

[0114] 另外,创建组件与上述创建快速开发页面的功能类似,不同点在于,创建组件时不自动生成路由,默认生成路径为组件component。

[0115] 以上为本申请实施例提供了一种前端脚手架的应用方法,该方法应用于通用前端脚手架,该方法可以包括:当通用前端脚手架接收到创建指令时,可以基于通用前端脚手架提供的前端框架选项,从中选择出目标前端框架。然后,可以基于通用前端脚手架提供的基础工程选项,从中确定出目标基础工程选项。选项映射地图中包括不同前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系。因此,可以在确定出目标前端框架和目标基础工程选项后,可以根据目标前端框架和目标基础工程选项,从通用前端脚手架提供的选项映射地图中,确定出用于通过目标基础工程选项实现的基础工程的代码存储地址,即目标基础工程地址。之后,可以通过下载代码指令,从映射出的目标基础工程地址中,拉取到目标基础工程,从而可以基于目标基础工程,创建前端页面。由此,通过通用前端脚手

架完成了项目初始化,可以基于通用前端脚手架提供的目标基础工程,继续进行前端开发。

[0116] 可见,通用前端脚手架中提供了选项映射地图,其中包含有各个前端框架和不同基础工程选项,与目标基础工程地址的映射关系,使得通用前端脚手架提供的基础工程可以适用于多种前端框架,并且能够通过目标基础工程选项实现对应的基础工程,从而基于目标基础工程可以继续前端开发,无需安装多种前端框架对应的脚手架,进而提升了开发效率。

[0117] 以上为本申请实施例提供前端脚手架的应用方法的一些具体实现方式,基于此,本申请还提供了对应的装置。下面将从功能模块化的角度对本申请实施例提供的装置进行介绍。

[0118] 参见图4,该图为本申请实施例提供的一种前端脚手架的应用装置400的结构示意图,该装置部署于通用前端脚手架,该装置400可以包括:

[0119] 第一确定模块401,用于当所述通用前端脚手架接收到创建指令时,基于所述通用前端脚手架提供的前端框架选项,确定出目标前端框架;

[0120] 第二确定模块402,用于基于所述通用前端脚手架提供的基础工程选项,确定出目标基础工程选项;

[0121] 第三确定模块403,用于基于所述目标前端框架和所述目标基础工程选项,从所述通用前端脚手架提供的选项映射地图中,确定出目标基础工程地址;所述目标基础功能地址用于指示通过所述目标基础工程选项实现的基础功能的代码存储地址;所述选项映射地图中包括不同前端框架和不同基础工程选项,与不同基础工程地址之间的映射关系;

[0122] 拉取模块404,用于通过下载代码指令,从所述目标基础工程地址,拉取目标基础工程;

[0123] 创建模块405,用于基于所述目标基础工程,创建前端页面。

[0124] 可选地,所述基础工程选项包括构建工具、用户界面组件库、路由封装选项、请求服务封装选项以及数据管理库封装选项。

[0125] 可选地,所述前端页面包括快速开发页面,所述创建模块405,包括:

[0126] 创建单元,用于基于所述目标基础工程,通过创建页面命令中的预设参数,创建快速开发页面,所述快速开发页面是由所述通用前端脚手架提供的页面模式。

[0127] 可选地,所述第一创建单元包括:

[0128] 确定单元,用于基于所述目标前端框架,从所述通用前端脚手架提供的页面模板映射地图中,确定出所述快速开发页面对应的快速开发页面模板;所述页面模板映射地图中包括不同前端框架与不同页面模板之间的映射关系;所述快速开发页面模板中包括表格组件和表单组件,以及表格组件和表单组件分别对应的基础交互操作;

[0129] 获得单元,用于基于所述快速开发页面模板和结构化数据配置文件,得到所述快速开发页面。

[0130] 可选地,所述创建页面命令中包括所述快速开发页面的路由地址,所述装置还包括:

[0131] 生成单元,用于通过基于所述快速开发页面模板和所述快速开发页面的路由地址,生成对应的代码文件,将所述代码文件存储于所述目标基础工程对应的文件夹中。

[0132] 可选地,所述装置还包括:

[0133] 读取单元,用于通过获取命令从所述目标基础工程对应的文件夹中读取存储文件,基于所述目标基础工程为所述存储文件划分存储位置。

[0134] 可选地,所述装置还包括:

[0135] 启动单元,用于通过命令行指令启动子进程,所述子进程用于安装所述目标基础工程的相关依赖和启动服务器。

[0136] 本申请实施例还提供了对应的设备以及计算机存储介质,用于实现本申请实施例提供的方案。

[0137] 其中,所述设备包括存储器和处理器,所述存储器用于存储计算机程序,所述处理器用于执行所述计算机程序,以使所述设备执行本申请任一实施例所述的前端脚手架的应用方法。

[0138] 所述计算机存储介质中存储有计算机程序,当所述代码被运行时,运行所述计算机程序的设备实现本申请任一实施例所述的前端脚手架的应用方法。

[0139] 本申请实施例中提到的“第一”、“第二”(若存在)等名称中的“第一”、“第二”只是用来做名字标识,并不代表顺序上的第一、第二。

[0140] 通过以上的实施方式的描述可知,本领域的技术人员可以清楚地了解到上述实施例方法中的全部或部分步骤可借助软件加通用硬件平台的方式来实现。基于这样的理解,本申请的技术方案可以以软件产品的形式体现出来,该计算机软件产品可以存储在存储介质中,如只读存储器(英文:read-only memory,ROM)/RAM、磁碟、光盘等,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者诸如路由器等网络通信设备)执行本申请各个实施例或者实施例的某些部分所述的方法。

[0141] 需要说明的是,本说明书中的各个实施例均采用递进的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于装置实施例而言,由于其基本相似于方法实施例,所以描述得比较简单,相关之处参见方法实施例的部分说明即可。以上所描述的装置实施例仅仅是示意性的,其中作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元提示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部模块来实现本实施例方案的目的。本领域普通技术人员在不付出创造性劳动的情况下,即可以理解并实施。

[0142] 以上所述,仅为本申请的一种具体实施方式,但本申请的保护范围并不局限于此,任何熟悉本技术领域的技术人员在本申请揭露的技术范围内,可轻易想到的变化或替换,都应涵盖在本申请的保护范围之内。因此,本申请的保护范围应该以权利要求的保护范围为准。

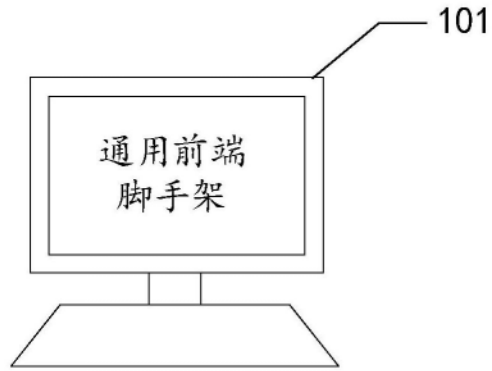


图1

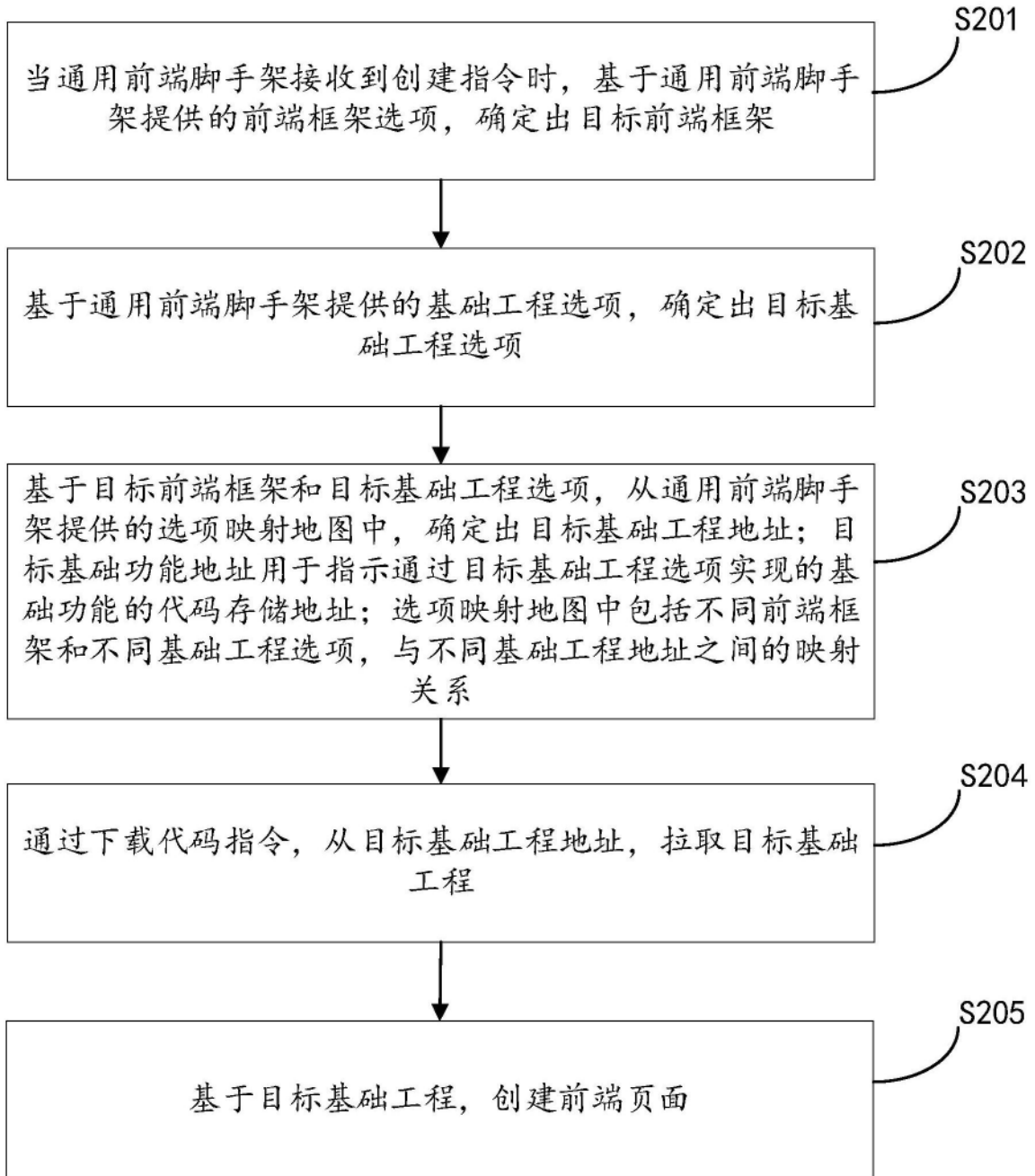


图2

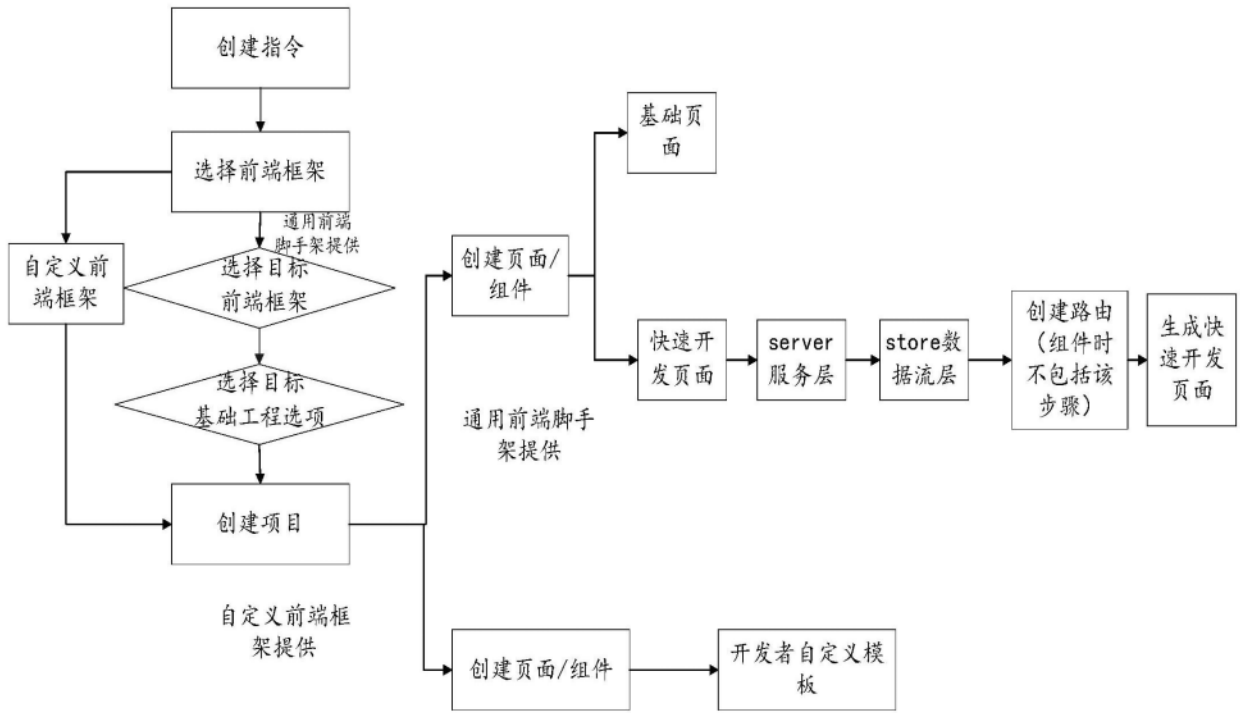


图3

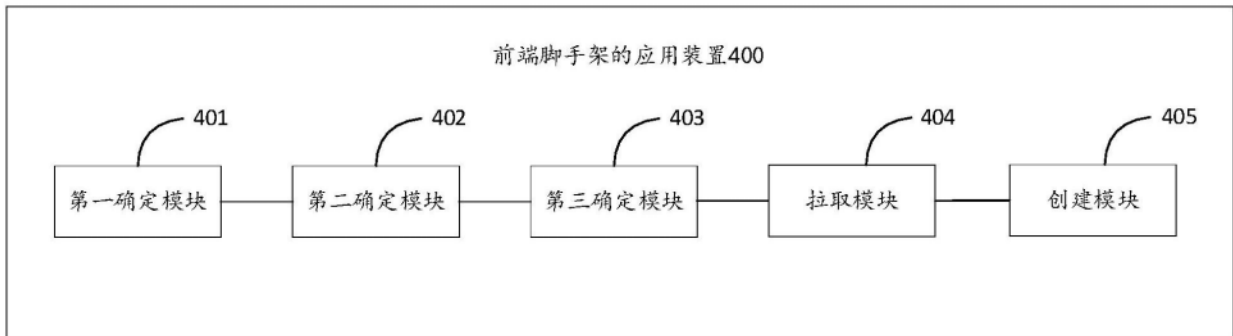


图4