

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2025 年 1 月 9 日 (09.01.2025)



(10) 国际公布号
WO 2025/007532 A1

(51) 国际专利分类号:
G06F 8/20 (2018.01)

(21) 国际申请号: PCT/CN2023/142715

(22) 国际申请日: 2023 年 12 月 28 日 (28.12.2023)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
202311712621.8 2023 年 12 月 13 日 (13.12.2023) CN

(71) 申请人: 天翼云科技有限公司 (CHINA TELECOM CLOUD TECHNOLOGY CO., LTD) [CN/CN]; 中国北京市东城区青龙胡同甲 1 号、3 号 2 幢 2 层 205-32 室, Beijing 100007 (CN)。

(72) 发明人: 周朝 (ZHOU, Chao); 中国北京市东城区青龙胡同甲 1 号、3 号 2 幢 2 层 205-32 室, Beijing 100007 (CN)。

(74) 代理人: 北京知汇林知识产权代理事务所 (普通合伙) (BEIJING ZHIHUILIN INTELLECTUAL PROPERTY LAW FIRM); 中国北京市海淀区海淀南路甲 21 号中关村知识产权大厦 A 座 7 层, Beijing 100080 (CN)。

(81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA,

(54) Title: METHOD AND SYSTEM FOR INITIALIZING CUSTOMIZABLE JAVA ENGINEERING

(54) 发明名称: 一种可定制化 Java 工程初始化方法及系统

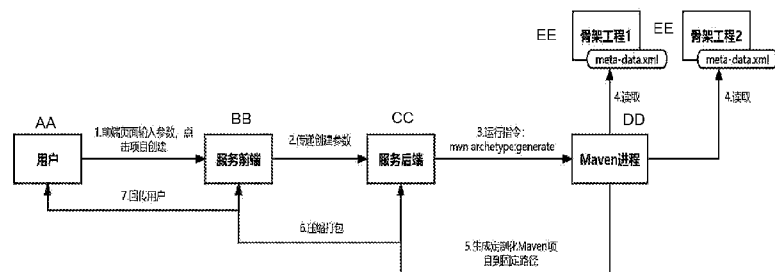


图 2

- 1 Input parameters on a front-end page, and click on project creation
 - 2 Transmit creation parameters
 - 3 Run an instruction: mvn archetype:generate
 - 4 Read
 - 5 Generate a customized Maven project to a fixed path
 - 6 Compress and package
 - 7 Transmit back to the user
- AA User
BB Service front end
CC Service back end
DD Maven process
EE Skeleton engineering

(57) Abstract: Disclosed in the present invention are a method and system for initializing customizable Java engineering. The method comprises: by means of a front-end service module, acquiring a creation request that is used for a front-end user to create a project, wherein the creation request for creating the project is created by means of the front-end user inputting page parameters and clicking; the front-end service module sending to a back-end service module the creation request for creating the project; the back-end service module calling a Maven process module to read a meta-data.xml file in a skeleton engineering module, so as to acquire a project creation rule, and then the Maven process module completing the creation of a customized Maven project, and generating template engineering; the Maven process module generating the customized Maven project to a fixed path, wherein the fixed path is from the Maven process module to the back-end service module; and the back-end service module compressing and packaging the customized Maven project,

PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW。

- (84) 指定国(除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

- 包括国际检索报告(条约第21条(3))。
- 在修改权利要求的期限届满之前进行, 在收到该修改后将重新公布(细则48.2(h))。
- 根据申请人的请求, 在条约第21条(2)(a)所规定的期限届满之前进行。

and then sending same to a user module. The method in the present invention can quickly initialize SpringBoot engineering, and can greatly improve the project creation efficiency of developers and reduce the dependence maintenance costs by means of importing related dependencies, solving dependency conflicts and generating various usage instructions.

(57) 摘要: 本发明公开了一种可定制化Java工程初始化方法及系统, 方法包括: 通过前端服务模块获取前端用户用于创建项目的创建请求, 创建项目的创建请求通过前端用户输入页面参数点击创建; 前端服务模块将创建项目的创建请求发送到后端服务模块, 后端服务模块调用Maven进程模块读取骨架工程模块中meta-data.xml文件获取项目创建规则后, Maven进程模块完成定制化Maven项目创建, 生成模板工程; Maven进程模块生成定制化Maven项目到固定路径, 固定路径是由Maven进程模块到后端服务模块, 后端服务模块将定制化Maven项目压缩打包后发送给用户模块, 本发明方法能够快速初始化SpringBoot工程, 通过导入相关依赖、解决依赖冲突、生成各类用法说明的方式, 极大提高开发者的项目创建效率、降低依赖维护成本。

一种可定制化 Java 工程初始化方法及系统

技术领域

本发明属于计算机技术领域，具体涉及一种可定制化 Java 工程初始化方法
5 及系统。

背景技术

软件工程的后端开发工作通常需要创建 Java 工程项目，也即代码工程的初始化。目前 SpringBoot 框架是 Java 后端的主流开发框架，其中包含大量实用的开发组件，这些开发组件以依赖的形式被 Maven 所管理，从而被导入到工程
10 中为工程提供相应能力。项目工程的初始化是所有开发人员开发新组件、新应用时无法绕过的一步，项目工程初始化的耗时、初始化后项目所呈现的形态，直接影响到整体项目的开发进度。目前，开发人员在项目初始化阶段存在以下的痛点：

1. 大量 SpringBoot 项目的首次创建均是基于空模板，各种组件依赖需要
15 依次手工导入；

2. 开发后期，依赖项较为复杂时，容易产生依赖版本的冲突，影响开发效率；

3. 依赖导入后，工程中仍缺乏所导入组件功能的用法、示例，同样影响开发效率。

20 对此，业界已经提出了很多 SpringBoot 项目快速生成和初始化的服务，这些服务可被免费在线使用。其中包括 Spring 官方的 Spring initializer，以及阿里云后续推出的 Aliyun Java Initializr，这些工具和服务都能在一定程度上加快开发者创建 Springboot 项目的速度，并在创建时提供可选项，根据开发者的需求，为创建后的项目工程添加对应的代码段，从而导入相关依
25 赖，为工程赋予对应的能力。然而，现存的项目生成服务存在以下的技术缺陷：

1. 很多项目生成服务只能提供项目名称、工程结构级别的定制，无法对具

体的开发组件和开发能力进行定制；

2. 很多项目生成服务无法为新创建的工程提供完整的用法示例和 Readme 文件；

3. 很多项目生成服务无法提供细粒度的代码段定制。

5 因此，导致生成的工程缺乏足够的可定制性，很多代码段需要开发者在后期手动补足，大大延长了开发时间。

发明内容

10 本发明的目的是提供一种可定制化 Java 工程初始化方法及系统，以解决现有技术中的不足，它通过首次提出了模板工程生成服务的概念，该服务能够快速初始化 **SpringBoot** 工程，通过导入相关依赖、解决依赖冲突、生成各类用法说明的方式，极大提高开发者的项目创建效率、降低依赖维护成本。

本申请的一个实施例提供了一种可定制化 Java 工程初始化方法包括：

15 通过前端服务模块获取前端用户用于创建项目的创建请求，所述创建项目的创建请求通过前端用户输入页面参数点击创建；

所述前端服务模块将创建项目的创建请求发送到后端服务模块，所述后端服务模块调用 **Maven** 进程模块读取骨架工程模块中 `meta-data.xml` 文件；

通过所述读取骨架工程模块中 `meta-data.xml` 文件获取项目创建规则后，所述 **Maven** 进程模块完成定制化 **Maven** 项目创建，生成模板工程；

20 所述 **Maven** 进程模块生成定制化 **Maven** 项目到固定路径，所述固定路径是由 **Maven** 进程模块到后端服务模块，所述后端服务模块将所述定制化 **Maven** 项目压缩打包后发送给用户模块。

可选的，所述 **Maven** 进程模块完成定制化 **Maven** 项目创建的步骤包括：

25 所述后端服务模块调用所述 **Maven** 进程模块执行 `archetype:generate` 指令；

所述 Maven 进程模块拉取本地中对应名称的骨架工程,找到所述骨架工程特定路径下的 archetype-metadata.xml 文件;

获取所述创建项目的生成配置,所述 Maven 进程模块开始执行所述创建项目的任务。

- 5 可选的,所述 Maven 进程模块开始执行所述创建项目的任务还包括读取骨架工程模块中的骨架 velocity 模板文件,渲染出定制化的代码工程文件。

可选的,所读取骨架工程模块中的骨架 velocity 模板文件,渲染出定制化的代码工程文件,包括:

替换所述骨架 velocity 模板文件中的相应字段;

- 10 根据输入参数对现有代码段进行增删,完成对应依赖导入;

所述根据输入参数对现有代码文件进行删除,删除不需要代码文件对应的模板文件。

可选的,所述骨架工程模块包含有若干组,所述模板工程提供所导入能力的基本用法示例。

- 15 可选的,所述方法还包括模版生成服务的搭建,具体步骤如下:

搭建前端服务,所述搭建前端服务为用户提供工程创建和可选项展示界面;

搭建后端服务,所述接收前端服务的创建请求,解析用户的定制参数项;

开发 maven archetype 骨架工程,所述开发 maven archetype 骨架工程完成后使用 mvn install 指令安装到本地 maven 仓库。

- 20 可选的,一种可定制化 Java 工程初始化系统。

本申请的又一实施例提供了一种可定制化 Java 工程初始化系统,所述系统包括:

用户模块,所述用户模块用于前端用户点击创建按钮,创建项目的创建请求

前端服务模块,所述前端服务模块用于传递创建参数给后端服务模块,并将所述定制化 Maven 项目压缩后返回给用户模块;

后端服务模块,所述后端服务模块用于接收所述前端服务模块创建项目的创建请求并调用 Maven 进程模块执行计算机指令;

5 Maven 进程模块,所述 Maven 进程模块用于读取骨架工程中的文件获取创建规则;

骨架工程模块,所述骨架工程模块用于通过创建参数创建项目框架。

本申请的又一实施例提供了一种存储介质,所述存储介质中存储有计算机程序,其中,所述计算机程序被设置为运行时实现上述任一项中所述的方法。

10 本申请的又一实施例提供了一种电子设备,包括存储器和处理器,所述存储器中存储有计算机程序,所述处理器被设置为运行所述计算机程序以实现上述任一项中所述的方法。

与现有技术相比,本文提出的 Java 工程初始化方法提供细粒度的工程能力融入:不但可以对包名、项目名进行定制,还可以对工程能力进行定制和融入(目前均是基于 Spring Boot,后期可支持其他框架);模板自动导入依赖并融入能力:免去手动添加依赖和必要配置项的麻烦,免解决依赖间的版本冲突,减少开发者对 pom 文件的维护时间,集中于业务开发;模板可提供基本示例代码及文档:提供所导入能力的基本用法示例,Readme 中附用法说明和官网文档链接等,减少开发者搜索用法和自测的时间,提升开发效率;更适用于
15 微服务应用搭建:针对微服务应用的搭建,提供更多实用的可选项,更适合与微服务平台相关产品接轨。例如,PAAS 平台的应用托管相关产品,在对应用开发、部署、监控、运维的整个生命周期流程管理中,现缺少一个能够快速生成应用的第一环节功能,本文提出的工程初始化方法即可补足该环节能力,提高相关产品的行业竞争力。

25

附图说明

图 1 为本发明实施例提供的一种可定制化 Java 工程初始化系统框图；

图 2 为本发明实施例提供的一种可定制化 Java 工程初始化方法的流程示意图；

图 3 为本发明实施例提供的模板工程生成服务骨架工程代码结构示例图；

5 图 4 为本发明实施例提供的使用可定制化 Java 工程初始化方法生成的一个 Java 工程代码结构示例图。

具体实施方式

下面通过参考附图描述的实施例是示例性的，仅用于解释本发明，而不能
10 解释为对本发明的限制。

本发明实施例首先提供了一种可定制化 Java 工程初始化方法，该方法可以应用于电子设备，如计算机终端，具体如普通电脑、量子计算机等。

参见图 2，图 2 为本发明实施例提供的一种可定制化 Java 工程初始化方法的流程示意图，可以包括如下步骤：

15 S1：通过前端服务模块获取前端用户用于创建项目的创建请求，所述创建项目的创建请求通过前端用户输入页面参数点击创建；

S2：所述前端服务模块将创建项目的创建请求发送到后端服务模块，所述后端服务模块调用 Maven 进程模块读取骨架工程模块中 meta-data.xml 文件；

20 S3：通过所述读取骨架工程模块中 meta-data.xml 文件获取项目创建规则后，所述 Maven 进程模块完成定制化 Maven 项目创建，生成模板工程；

S4：所述 Maven 进程模块生成定制化 Maven 项目到固定路径，所述固定路径是由 Maven 进程模块到后端服务模块，所述后端服务模块将所述定制化 Maven 项目压缩打包后发送给用户模块。

具体的，模板工程生成服务为用户提供友好的项目生成和订制界面，前端
25 用户点击创建按钮，项目创建请求发出后，后端调用 maven 进程执行 mvn

`archetype:generate` 指令，maven 进程从本地拉取对应的骨架工程，通过读取骨架工程中的 `meta-data.xml` 文件获取创建规则，完成定制化项目创建，后端将项目压缩后返回给用户，供用户下载使用。

其中，由骨架工程生成定制化项目的流程细节如下：

- 5 1、Maven 进程接收到 `archetype:generate` 指令；
- 2、Maven 进程拉取本地中对应名称的骨架工程，找到骨架工程特定路径下的 `archetype-metadata.xml` 文件，获取项目生成配置；
- 3、Maven 进程开始执行项目创建任务，读取骨架工程中的骨架 `velocity` 模板文件，渲染出定制化的代码工程文件。

- 10 在一种可选的实施方式中，Maven 进程开始执行项目创建任务，读取骨架工程中的骨架 `velocity` 模板文件，渲染出定制化的代码工程文件，其中包括：

替换模板中的相应字段，如项目名称、机构、作者、工程服务名等字段；

根据输入参数对现有代码段进行增删，如修改 `pom.xml` 文件，完成对应依赖导入；

- 15 根据输入参数对现有代码文件进行删除，删除不需要的模板文件，仅留下用户所需依赖能力的对应代码类、示例代码和说明文件等。

进一步的，模板工程生成服务的具体搭建步骤如下：

步骤 1、搭建前端服务，为用户提供友好的工程创建和可选项展示界面，最终呈现的界面效果如图 2 所示；

- 20 步骤 2、搭建后端服务，接收前端创建请求，解析用户定制参数项，即时调用 maven 进程执行带参数的 `mvn archetype:generate` 指令。最终的拼装指令类如下：

```
mvn archetype:generate
```

```
"-DgroupId=com. zhouzhao"
```

- 25

```
"-DartifactId=test-from-archetype-v1"
```

```
"-Dversion=2.0.0-SNAPSHOT"

"-Dauthor=zhouzhao"

"-Dpackage=com.zhouzhao.test.from.archetype.v1"

"-DarchetypeGroupId=com.ctg.mse.platform"

5  "-DarchetypeArtifactId=svc-api"

"-DarchetypeVersion=1.2.0-SNAPSHOT"

"-DarchetypeCatalog=local"

"-DinteractiveMode=false"

"-Dredis=false"

10 "-Dmysql=false";
```

步骤3、开发 maven archetype 骨架工程，主要涉及定制参数的接收和处理、项目工程结构设计、velocity 模板文件编写、生成测试等环节。骨架工程开发完成后，使用 mvn install 指令安装到本地 maven 仓库。所开发的骨架工程代码结构类如图3所示。

15 通过以上步骤，即可开发一套完整的模板工程生成服务。完成部署后，使用此服务即可实现 Java 工程的高度定制化。使用该初始化方法生成的 Java 工程可以快捷地用于业务增量开发，一个所生成的 Java 工程的目录结构示例如图4所示，根据用户的需求，该工程被生成为包含 MySQL 能力以及 Nacos 注册中心能力，MySQL 相关代码会被生成在图4中 java 目录的 dao 包中，Spring
20 相关依赖会被自动导入（pom 文件被改写），相关配置会被写入 application.yml 文件，如图5所示。

本申请的又一实施例提供了一种可定制化 Java 工程初始化系统，如图1所示的一种可定制化 Java 工程初始化系统的模块示意图，所述系统包括：

用户模块，所述用户模块用于前端用户点击创建按钮，创建项目的创建请
25 求

前端服务模块,所述前端服务模块用于传递创建参数给后端服务模块,并将所述定制化 Maven 项目压缩后返回给用户模块;

后端服务模块,所述后端服务模块用于接收所述前端服务模块创建项目的创建请求并调用 Maven 进程模块执行计算机指令;

5 Maven 进程模块,所述 Maven 进程模块用于读取骨架工程中的文件获取创建规则;

骨架工程模块,所述骨架工程模块用于通过创建参数创建项目框架。

具体的, Maven 进程模块通过坐标的方式获取 Jar 包,坐标的组成为:公司/组织 (groupId)+项目名 (artifactId)+版本 (version) 组成,可以从
10 互联网、本地等多种仓库源获取 Jar 包。

具体的,后端服务模块创建项目完成后,需要导入所需的 jar,要通过【坐标】来导入,其中每个构件都有自己的坐标(标识) = groupId + artifactId + version = 项目标识 + 项目名 + 版本,在定制化 Maven 项目中只需要配置坐标, maven 便会自动加载对应依赖,删除坐标则会移除依赖,同时 jar 包并不会导入到项目目录中,只是 Maven 进程模块帮助做了关联。但在打包部署项目
15 时会携带对应依赖,如果导入了依赖 A,而 A 中又依赖了 B,则 B 会被自动加载。

与现有技术相比,本发明中不但可以对包名、项目名进行定制,还可以对工程能力进行定制和融入;而且模板自动导入依赖并融入能力:免去手动添加
20 依赖和必要配置项的麻烦,免解决依赖间的版本冲突,减少开发者对 pom 文件的维护时间,集中于业务开发,同时本发明中的模板可提供基本示例代码及文档:提供所导入能力的基本用法示例, Readme 中附用法说明和官网文档链接等,减少开发者搜索用法和自测的时间,提升开发效率,本发明更适用于微服务应用搭建:针对微服务应用的搭建,提供更多实用的可选项,更适合与微服务平台相关产品接轨。例如, PAAS 平台的应用托管相关产品,在对应用开发、
25 部署、监控、运维的整个生命周期流程管理中,现缺少一个能够快速生成应用的第一环节功能,本文提出的工程初始化方法即可补足该环节能力,提高相关

产品的行业竞争力。

本发明实施例还提供了一种存储介质，所述存储介质中存储有计算机程序，其中，所述计算机程序被设置为运行时实现上述任一项方法实施例中的步骤。

具体的，在本实施例中，上述存储介质可以被设置为存储用于执行以下步骤的计算机程序：

S1：通过前端服务模块获取前端用户用于创建项目的创建请求，所述创建项目的创建请求通过前端用户输入页面参数点击创建；

S2：所述前端服务模块将创建项目的创建请求发送到后端服务模块，所述后端服务模块调用 Maven 进程模块读取骨架工程模块中 meta-data.xml 文件；

10 S3：通过所述读取骨架工程模块中 meta-data.xml 文件获取项目创建规则后，所述 Maven 进程模块完成定制化 Maven 项目创建，生成模板工程；

S4：所述 Maven 进程模块生成定制化 Maven 项目到固定路径，所述固定路径是由 Maven 进程模块到后端服务模块，所述后端服务模块将所述定制化 Maven 项目压缩打包后发送给用户模块。

15 具体的，在本实施例中，上述存储介质可以包括但不限于：U 盘、只读存储器(Read-Only Memory, 简称为 ROM)、随机存取存储器(Random Access Memory, 简称为 RAM)、移动硬盘、磁碟或者光盘等各种可以存储计算机程序的介质。

与现有技术相比，本文提出的 Java 工程初始化方法提供细粒度的工程能力融入：不但可以对包名、项目名进行定制，还可以对工程能力进行定制和融入（目前均是基于 Spring Boot，后期可支持其他框架）；模板自动导入依赖并融入能力：免去手动添加依赖和必要配置项的麻烦，免解决依赖间的版本冲突，减少开发者对 pom 文件的维护时间，集中于业务开发；模板可提供基本示例代码及文档：提供所导入能力的基本用法示例，Readme 中附用法说明和官网文档链接等，减少开发者搜索用法和自测的时间，提升开发效率；更适用于
20 微服务应用搭建：针对微服务应用的搭建，提供更多实用的可选项，更适合与微服务平台相关产品接轨。例如，PAAS 平台的应用托管相关产品，在对应用开发、部署、监控、运维的整个生命周期流程管理中，现缺少一个能够快速生

成应用的第一环节功能，本文提出的工程初始化方法即可补足该环节能力，提高相关产品的行业竞争力。

5 本发明实施例还提供了一种电子设备，包括存储器和处理器，所述存储器中存储有计算机程序，所述处理器被设置为运行所述计算机程序以执行上述任一项方法实施例中的步骤。

具体的，上述电子设备还可以包括传输设备以及输入输出设备，其中，该传输设备和上述处理器连接，该输入输出设备和上述处理器连接。

具体的，在本实施例中，上述处理器可以被设置为通过计算机程序执行以下步骤：

10 S1：通过前端服务模块获取前端用户用于创建项目的创建请求，所述创建项目的创建请求通过前端用户输入页面参数点击创建；

S2：所述前端服务模块将创建项目的创建请求发送到后端服务模块，所述后端服务模块调用 Maven 进程模块读取骨架工程模块中 meta-data.xml 文件；

15 S3：通过所述读取骨架工程模块中 meta-data.xml 文件获取项目创建规则后，所述 Maven 进程模块完成定制化 Maven 项目创建，生成模板工程；

S4：所述 Maven 进程模块生成定制化 Maven 项目到固定路径，所述固定路径是由 Maven 进程模块到后端服务模块，所述后端服务模块将所述定制化 Maven 项目压缩打包后发送给用户模块。

20 与现有技术相比，本文提出的 Java 工程初始化方法提供细粒度的工程能力融入：不但可以对包名、项目名进行定制，还可以对工程能力进行定制和融入（目前均是基于 Spring Boot，后期可支持其他框架）；模板自动导入依赖并融入能力：免去手动添加依赖和必要配置项的麻烦，免解决依赖间的版本冲突，减少开发者对 pom 文件的维护时间，集中于业务开发；模板可提供基本示例代码及文档：提供所导入能力的基本用法示例，Readme 中附用法说明和官网文档链接等，减少开发者搜索用法和自测的时间，提升开发效率；更适用于微服务应用搭建：针对微服务应用的搭建，提供更多实用的可选项，更适合与微服务平台相关产品接轨。例如，PAAS 平台的应用托管相关产品，在对应用

开发、部署、监控、运维的整个生命周期流程管理中，现缺少一个能够快速生成应用的第一环节功能，本文提出的工程初始化方法即可补足该环节能力，提高相关产品的行业竞争力。

需要说明的是，对于前述的各方法实施例，为了简单描述，故将其都表述为一系列的动作组合，但是本领域技术人员应该知悉，本发明并不受所描述的动作顺序的限制，因为依据本发明，某些步骤可以采用其他顺序或者同时进行。其次，本领域技术人员也应该知悉，说明书中所描述的实施例均属于优选实施例，所涉及的动作和模块并不一定是本发明所必须的。

在上述实施例中，对各个实施例的描述都各有侧重，某个实施例中沒有详述的部分，可以参见其他实施例的相关描述。

在本发明所提供的几个实施例中，应该理解到，所揭露的装置，可通过其它的方式实现。例如，以上所描述的装置实施例仅仅是示意性的，例如上述单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个系统，或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口，装置或单元的间接耦合或通信连接，可以是电性或其它的形式。

上述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，作为单元显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单元来实现本实施例方案的目的。

另外，在本发明各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现，也可以采用软件功能单元的形式实现。

上述集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用时，可以存储在一个计算机可读取存储器中。基于这样的理解，本发明

的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的全部或部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储器中,包括若干指令用以使得一台计算机设备(可为个人计算机、服务器或者网络设备等)执行本发明各个实施例上述方法的全部或部分步骤。而前述的存储器包括: U 盘、只读存储器 (ROM, Read-Only Memory)、随机存取存储器 (RAM, Random Access Memory)、移动硬盘、磁碟或者光盘等各种可以存储程序代码的介质。

以上对本发明实施例进行了详细介绍,本文中应用了具体个例对本发明的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本发明的方法及其核心思想;同时,对于本领域的一般技术人员,依据本发明的思想,在具体实施方式及应用范围上均会有改变之处,综上所述,本说明书内容不应理解为对本发明的限制。

权利要求书

1. 一种可定制化 Java 工程初始化方法，其特征在于，所述方法包括：
通过前端服务模块获取前端用户用于创建项目的创建请求，所述创建项目的创建请求通过前端用户输入页面参数点击创建；

5 所述前端服务模块将创建项目的创建请求发送到后端服务模块，所述后端服务模块调用 Maven 进程模块读取骨架工程模块中 meta-data.xml 文件；

通过所述读取骨架工程模块中 meta-data.xml 文件获取项目创建规则后，所述 Maven 进程模块完成定制化 Maven 项目创建，生成模板工程；

10 所述 Maven 进程模块生成定制化 Maven 项目到固定路径，所述固定路径是由 Maven 进程模块到后端服务模块，所述后端服务模块将所述定制化 Maven 项目压缩打包后发送给用户模块。

2. 根据权利要求 1 所述的一种可定制化 Java 工程初始化方法，其特征在于，所述 Maven 进程模块完成定制化 Maven 项目创建的步骤包括：

15 所述后端服务模块调用所述 Maven 进程模块执行 archetype:generate 指令；

所述 Maven 进程模块拉取本地中对应名称的骨架工程，找到所述骨架工程特定路径下的 archetype-metadata.xml 文件；

获取所述创建项目的生成配置，所述 Maven 进程模块开始执行所述创建项目的任务。

20 3. 根据权利要求 2 所述的一种可定制化 Java 工程初始化方法，其特征在于，所述 Maven 进程模块开始执行所述创建项目的任务还包括读取骨架工程模块中的骨架 velocity 模板文件，渲染出定制化的代码工程文件。

25 4. 根据权利要求 3 所述的一种可定制化 Java 工程初始化方法，其特征在于，所述读取骨架工程模块中的骨架 velocity 模板文件，渲染出定制化的代码工程文件，包括：

替换所述骨架 velocity 模板文件中的相应字段；

根据输入参数对现有代码段进行增删，完成对应依赖导入；

所述根据输入参数对现有代码文件进行删除，删除不需要代码文件对应的模板文件。

5. 根据权利要求 4 所述的一种可定制化 Java 工程初始化方法，其特征在于，所述骨架工程模块包含有若干组，所述模板工程提供所导入能力的基本用法示例。

6. 根据权利要求 5 所述的一种可定制化 Java 工程初始化方法，其特征在于，所述方法还包括模版生成服务的搭建，具体步骤如下：

搭建前端服务，所述搭建前端服务为用户提供工程创建和可选项展示界面；

搭建后端服务，所述接收前端服务的创建请求，解析用户的定制参数项；

开发 maven archetype 骨架工程，所述开发 maven archetype 骨架工程完成后使用 mvn install 指令安装到本地 maven 仓库。

7. 根据权利要求 6 所述的一种可定制化 Java 工程初始化方法，其特征在于，所述开发 maven archetype 骨架工程包括：

用户定制参数的接收和处理，用于接收处理所述解析用户的定制参数项；

项目工程结构设计，对不同的场景去抽象不同的模块，通过通过依赖关系对项目工程结构进行设计对骨架 velocity 模板文件编写；

15 maven archetype 骨架工程生成测试环节，使用 Maven 命令或者集成开发环境中的测试插件执行测试用例，并生成测试报告。

8. 一种可定制化 Java 工程初始化系统，其特征在于，所述系统包括：

用户模块，所述用户模块用于前端用户点击创建按钮，创建项目的创建请求

20 前端服务模块，所述前端服务模块用于传递创建参数给后端服务模块，并将所述定制化 Maven 项目压缩后返回给用户模块；

后端服务模块，所述后端服务模块用于接收所述前端服务模块创建项目的创建请求并调用 Maven 进程模块执行计算机指令；

25 Maven 进程模块，所述 Maven 进程模块用于读取骨架工程中的文件获取创建规则；

骨架工程模块，所述骨架工程模块用于通过创建参数创建项目框架。

9. 一种存储介质，其特征在于，所述存储介质中存储有计算机程序，其中，所述计算机程序被设置为运行时实现所述权利要求 1 至 7 任一项中所述的方法。

10. 一种电子设备，包括存储器和处理器，其特征在于，所述存储器中存

储有计算机程序,所述处理器被设置为运行所述计算机程序以实现所述权利要求 1 至 7 任一项中所述的方法。

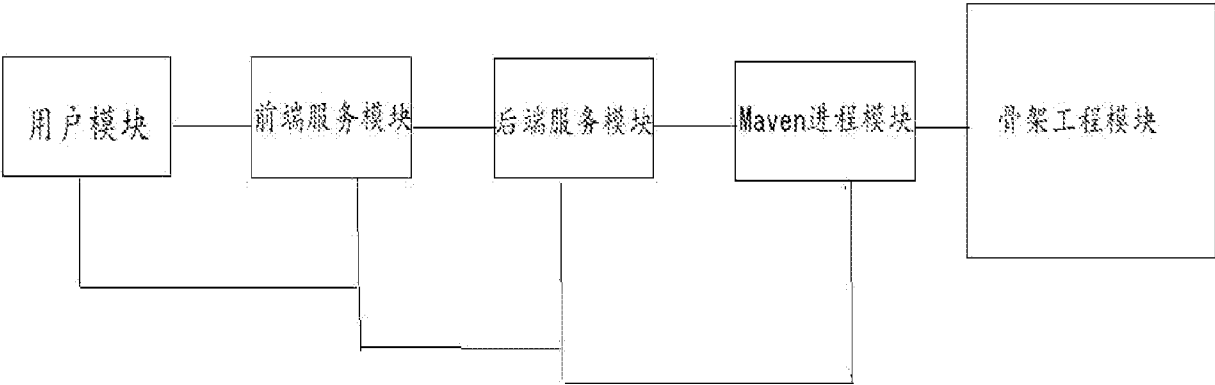


图 1

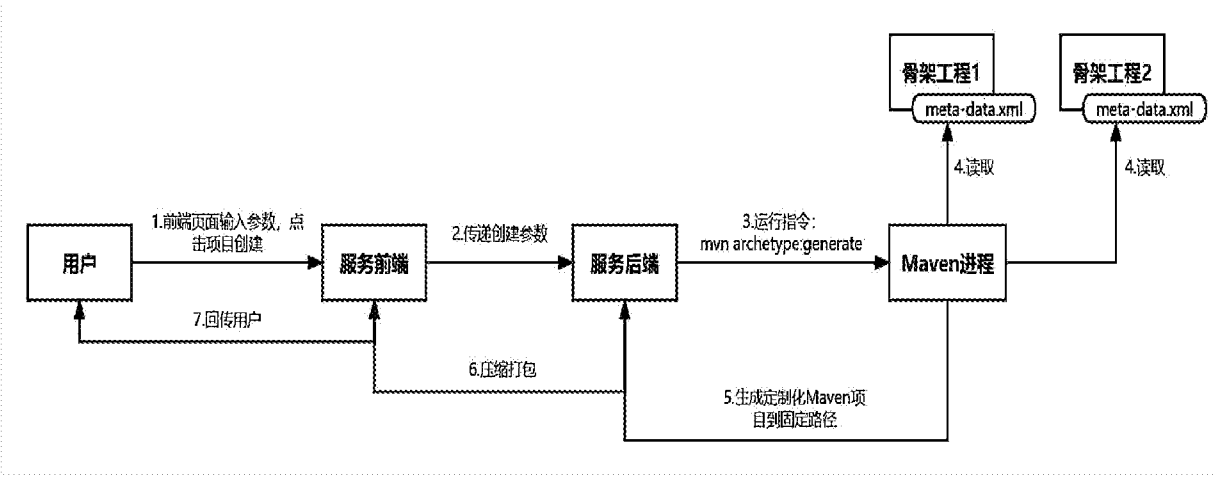


图 2

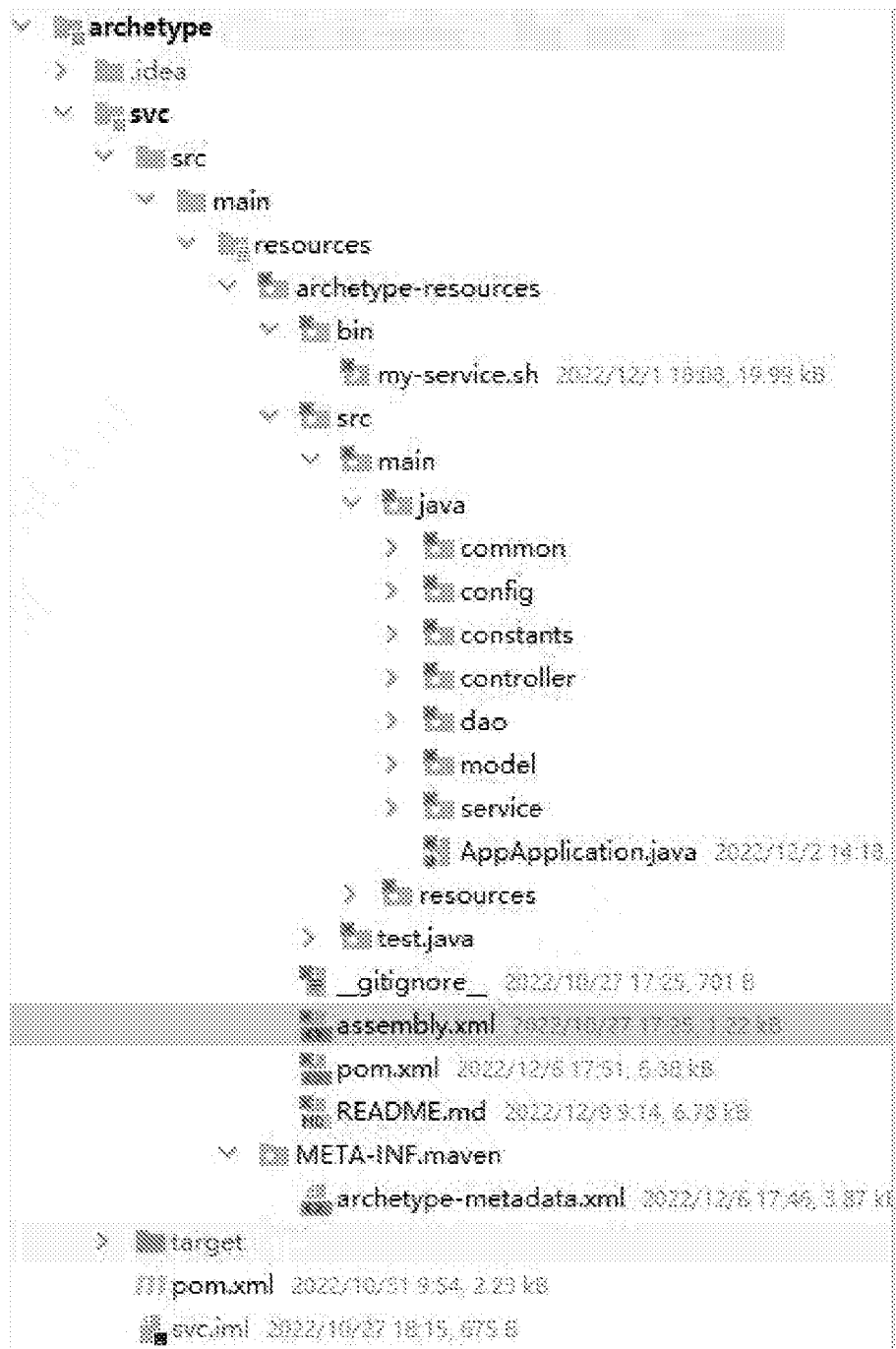


图 3

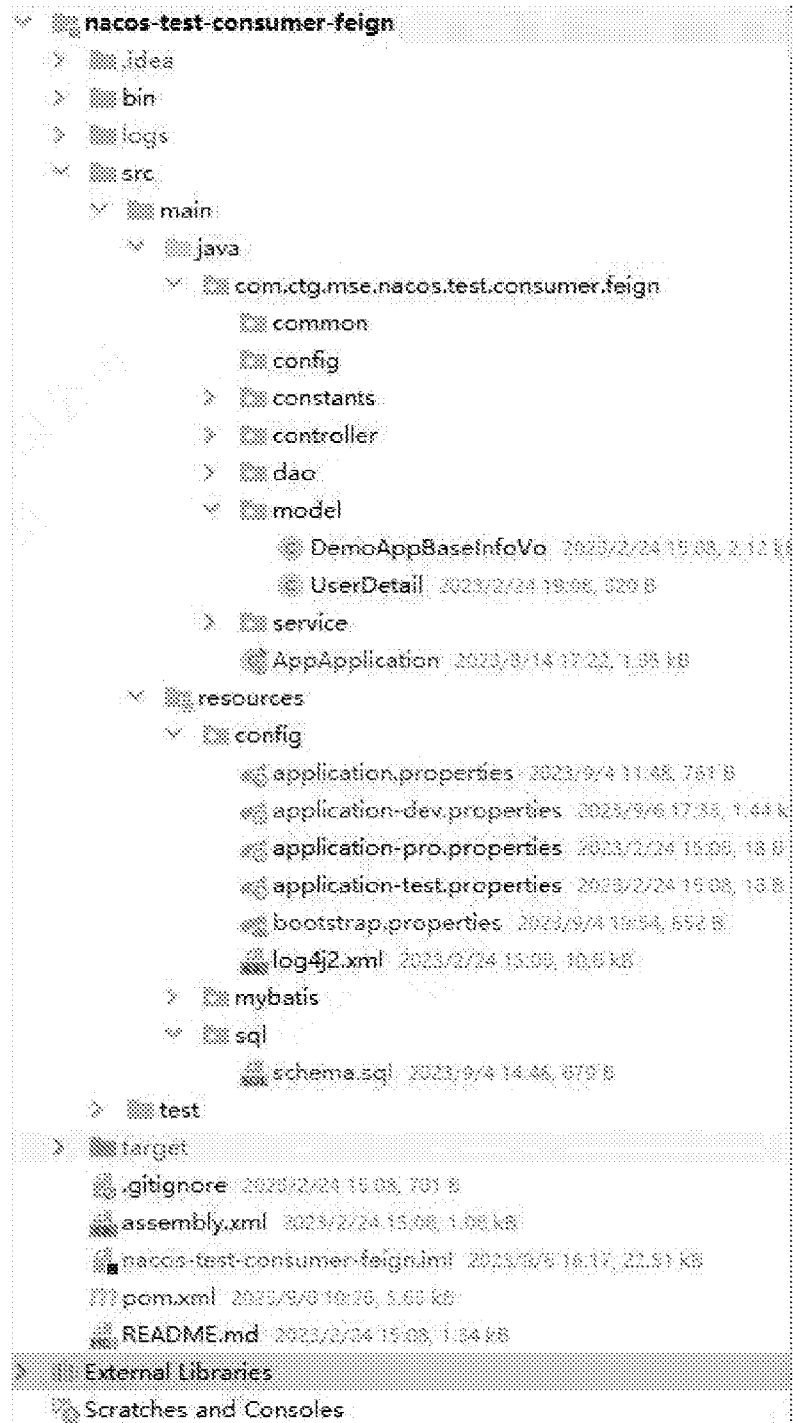


图 4

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2023/142715

A. CLASSIFICATION OF SUBJECT MATTER

G06F 8/20(2018.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

VEN, CNABS, CNTXT, WOTXT, EPTXT, USTXT, CNKI, IEEE: java, maven, 初始化, 模板, 定制, 骨架, 元数据, 脚手架, 路径, 参数, 代码, 工程, 项目; java, maven, initialization, template, customization, skeleton, metadata, scaffold+, path, parameter, code, project

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 113127098 A (KANGJIAN INFORMATION TECHNOLOGY (SHENZHEN) CO., LTD.) 16 July 2021 (2021-07-16) description, paragraphs 67-163, and figures 1-5	1-10
A	CN 114564175 A (SOUTH DIGITAL TECHNOLOGY CO., LTD.) 31 May 2022 (2022-05-31) entire document	1-10
A	CN 111290753 A (YGSOFT INC.) 16 June 2020 (2020-06-16) entire document	1-10
A	CN 113805877 A (ICBC TECHNOLOGY CO., LTD. et al.) 17 December 2021 (2021-12-17) entire document	1-10
A	CN 115658025 A (DALIAN JINCHENG NETWORK TECHNOLOGY CO., LTD.) 31 January 2023 (2023-01-31) entire document	1-10



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance
“D” document cited by the applicant in the international application
“E” earlier application or patent but published on or after the international filing date
“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
“O” document referring to an oral disclosure, use, exhibition or other means
“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search

30 June 2024

Date of mailing of the international search report

08 July 2024

Name and mailing address of the ISA/CN

China National Intellectual Property Administration (ISA/
CN)
China No. 6, Xitucheng Road, Jimenqiao, Haidian District,
Beijing 100088

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2023/142715

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN 115951879 A (BEIJING VOLCANO ENGINE TECHNOLOGY CO., LTD.) 11 April 2023 (2023-04-11) entire document	1-10
A	CN 117055858 A (GOODWELL TECHNOLOGY CO., LTD.) 14 November 2023 (2023-11-14) entire document	1-10

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2023/142715

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN	113127098	A	16 July 2021	None	
CN	114564175	A	31 May 2022	None	
CN	111290753	A	16 June 2020	None	
CN	113805877	A	17 December 2021	None	
CN	115658025	A	31 January 2023	None	
CN	115951879	A	11 April 2023	None	
CN	117055858	A	14 November 2023	None	

A. 主题的分类

G06F 8/20(2018.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

IPC: G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))

VEN, CNABS, CNTXT, WOTXT, EPTXT, USTXT, CNKI, IEEE: java, maven, 初始化, 模板, 定制, 骨架, 元数据, 脚手架, 路径, 参数, 代码, 工程, 项目; java, maven, initialization, template, customization, skeleton, metadata, scaffold+, path, parameter, code, project

C. 相关文件

类型*	引用文件, 必要时, 指明相关段落	相关的权利要求
X	CN 113127098 A (康健信息技术(深圳)有限公司) 2021年7月16日 (2021 - 07 - 16) 说明书第67-163段, 附图1-5	1-10
A	CN 114564175 A (广东南方数码科技股份有限公司) 2022年5月31日 (2022 - 05 - 31) 全文	1-10
A	CN 111290753 A (远光软件股份有限公司) 2020年6月16日 (2020 - 06 - 16) 全文	1-10
A	CN 113805877 A (工银科技有限公司 等) 2021年12月17日 (2021 - 12 - 17) 全文	1-10
A	CN 115658025 A (大连锦程网络技术有限公司) 2023年1月31日 (2023 - 01 - 31) 全文	1-10
A	CN 115951879 A (北京火山引擎科技有限公司) 2023年4月11日 (2023 - 04 - 11) 全文	1-10
A	CN 117055858 A (固德威技术股份有限公司) 2023年11月14日 (2023 - 11 - 14) 全文	1-10

☐ 其余文件在C栏的续页中列出。

☒ 见同族专利附件。

* 引用文件的具体类型:

“A” 认为不特别相关的表示了现有技术一般状态的文件

“D” 申请人在国际申请中引证的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件(如具体说明的)

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期

2024年6月30日

国际检索报告邮寄日期

2024年7月8日

ISA/CN的名称和邮寄地址

中国国家知识产权局
中国北京市海淀区蓟门桥西土城路6号 100088

授权官员

李福永

电话号码 (+86) 010-53961311

PCT/ISA/210 表(第2页) (2022年7月)

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2023/142715

检索报告引用的专利文件			公布日 (年/月/日)	同族专利	公布日 (年/月/日)
CN	113127098	A	2021年7月16日	无	
CN	114564175	A	2022年5月31日	无	
CN	111290753	A	2020年6月16日	无	
CN	113805877	A	2021年12月17日	无	
CN	115658025	A	2023年1月31日	无	
CN	115951879	A	2023年4月11日	无	
CN	117055858	A	2023年11月14日	无	