



(12) 发明专利申请

(10) 申请公布号 CN 112083926 A

(43) 申请公布日 2020.12.15

(21) 申请号 202010948715.5

(22) 申请日 2020.09.10

(71) 申请人 武汉开目信息技术股份有限公司
地址 430000 湖北省武汉市东湖新技术开发区花城大道9号武汉软件新城1.1期B3栋3楼

(72) 发明人 李盈 胡先兵 丁锐 邓立

(74) 专利代理机构 北京路浩知识产权代理有限公司 11002

代理人 苗晓静

(51) Int. Cl.

G06F 8/38 (2018.01)

G06F 16/958 (2019.01)

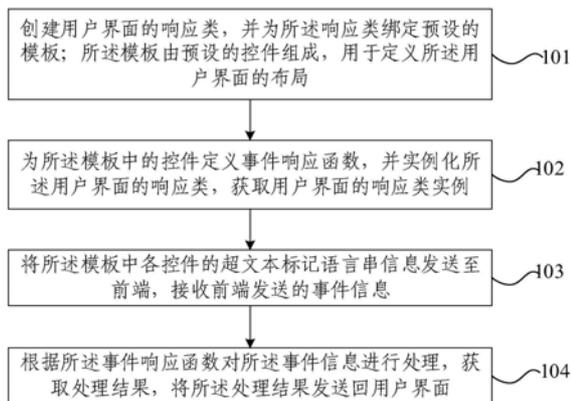
权利要求书2页 说明书8页 附图4页

(54) 发明名称

一种Web用户界面生成方法及装置

(57) 摘要

本发明实施例提供一种Web用户界面生成方法、装置、电子设备及存储介质,所述方法包括:创建用户界面的响应类,并为所述响应类绑定预设的模板;所述模板由预设的控件组成,用于定义所述用户界面的布局;为所述模板中的控件定义事件响应函数,并实例化所述用户界面的响应类,获取用户界面的响应类实例;将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。通过为响应类绑定模板,动态生成前端页面,实现事件与数据的绑定,后端提供业务响应,前端动态局部更新,提高了开发效率,降低了开发和维护的成本。



1. 一种Web用户界面生成方法,其特征在于,包括:

创建用户界面的响应类,并为所述响应类绑定预设的模板;所述模板由预设的控件组成,用于定义所述用户界面的布局;

为所述模板中的控件定义事件响应函数,并实例化所述用户界面的响应类,获取用户界面的响应类实例;

将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;

根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。

2. 根据权利要求1所述的Web用户界面生成方法,其特征在于,所述将所述模板中各控件的超文本标记语言串信息发送至前端,之前还包括:

对所述模板进行解析处理,分别获取所述模板中各控件的超文本标记语言串信息。

3. 根据权利要求2所述的Web用户界面生成方法,其特征在于,所述对所述模板进行解析处理,分别获取所述模板中各控件的超文本标记语言串信息,具体包括:

获取所述模板中各控件的控件定义;

基于所述控件定义获取所述模板中各控件的超文本标记语言串信息。

4. 根据权利要求1所述的Web用户界面生成方法,其特征在于,所述接收前端发送的事件信息,具体包括:

根据各个控件的ID获取各个控件对应的所述控件实例和所述用户界面的响应类实例;

通过所述用户界面的响应类实例绑定的事件接收函数接收所述事件信息。

5. 根据权利要求1所述的Web用户界面生成方法,其特征在于,所述根据所述事件响应函数对所述事件信息进行处理,具体包括:

调用所述事件响应函数;

根据所述事件信息中的各控件的事件类型和事件参数进行对应的处理。

6. 一种Web用户界面生成方法,其特征在于,包括:

接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局;

获取所述超文本标记语言串信息中的JS信息,并进行处理。

7. 一种Web用户界面生成装置,其特征在于,包括:

创建模块:用于创建用户界面的响应类,并为所述响应类绑定预设的模板;所述模板由预设的控件组成,用于定义所述用户界面的布局;

实例化模块,用于为所述模板中的控件定义事件响应函数,并实例化所述用户界面的响应类,获取用户界面的响应类实例;

交互模块,用于将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;

处理模块,用于根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。

8. 一种Web用户界面生成装置,其特征在于,包括:

接收模块,用于接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局;

处理模块,用于获取所述超文本标记语言串信息中的JS信息,并进行处理。

9.一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,其特征在于,所述处理器执行所述程序时实现如权利要求1至6任一项所述一种Web用户界面生成方法的步骤。

10.一种非暂态计算机可读存储介质,其上存储有计算机程序,其特征在于,该计算机程序被处理器执行时实现如权利要求1至6任一项所述一种Web用户界面生成方法的步骤。

一种Web用户界面生成方法及装置

技术领域

[0001] 本发明涉及互联网信息技术领域,尤其涉及一种Web用户界面生成方法及装置。

背景技术

[0002] 在工业软件中,由于软件界面内含有大量的实时数据与复杂的业务处理逻辑,采用常规的Web页面中很难实现且性能低下(前端页面通常采用的是JS脚本语言)。

[0003] 现行的开发方式通常采用前后端分离的方法进行开发,在前端编写每个页面的布局、数据绑定、业务处理等编码;在后端提供各种数据请求接口。

[0004] 现行的开发方式需要为每个页面编写其布局样式,页面布局偏静态,很难根据业务需要进行变形操作;绑定的数据量过大会降低页面响应性能;由于是弱语言,很难实现数据的高效计算与操作;业务处理功能代码与界面布局、数据绑定代码混淆在一起,难于维护扩展及复用,导致了开发与维护成本较高,且开发效率低下的缺陷。

发明内容

[0005] 本发明实施例提供一种Web用户界面生成方法、装置、电子设备及存储介质,用以解决现有技术中开发与维护成本较高,且开发效率低下的缺陷,实现开发效率的提高。

[0006] 本发明实施例提供一种Web用户界面生成方法,包括:

[0007] 创建用户界面的响应类,并为所述响应类绑定预设的模板;所述模板由预设的控件组成,用于定义所述用户界面的布局;

[0008] 为所述模板中的控件定义事件响应函数,并实例化所述用户界面的响应类,获取用户界面的响应类实例;

[0009] 将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;

[0010] 根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。

[0011] 根据本发明一个实施例的Web用户界面生成方法,所述将所述模板中各控件的超文本标记语言串信息发送至前端,之前还包括:

[0012] 对所述模板进行解析处理,分别获取所述模板中各控件的超文本标记语言串信息。

[0013] 根据本发明一个实施例的Web用户界面生成方法,所述对所述模板进行解析处理,分别获取所述模板中各控件的超文本标记语言串信息,具体包括:

[0014] 获取所述模板中各控件的控件定义;

[0015] 基于所述控件定义获取所述模板中各控件的超文本标记语言串信息。

[0016] 根据本发明一个实施例的Web用户界面生成方法,所述接收前端发送的事件信息,具体包括:

[0017] 根据各个控件的ID获取各个控件对应的所述控件实例和所述用户界面的响应类

实例；

[0018] 通过所述用户界面的响应类实例绑定的事件接收函数接收所述事件信息。

[0019] 根据本发明一个实施例的Web用户界面生成方法,所述根据所述事件响应函数对所述事件信息进行处理,具体包括:

[0020] 调用所述事件响应函数;

[0021] 根据所述事件信息中的各控件的事件类型和事件参数进行对应的处理。

[0022] 本发明实施例还提供一种Web用户界面生成方法,包括:

[0023] 接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局;

[0024] 获取所述超文本标记语言串信息中的JS信息,并进行处理。

[0025] 本发明实施例还提供一种Web用户界面生成装置,包括:

[0026] 创建模块:用于创建用户界面的响应类,并为所述响应类绑定预设的模板;所述模板由预设的控件组成,用于定义所述用户界面的布局;

[0027] 实例化模块,用于为所述模板中的控件定义事件响应函数,并实例化所述用户界面的响应类,获取用户界面的响应类实例;

[0028] 交互模块,用于将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;

[0029] 处理模块,用于根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。

[0030] 本发明实施例还提供一种Web用户界面生成装置,包括:

[0031] 接收模块,用于接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局;

[0032] 处理模块,用于获取所述超文本标记语言串信息中的JS信息,并进行处理。

[0033] 本发明实施例还提供一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,所述处理器执行所述程序时实现如上述任一种所述一种Web用户界面生成方法的步骤。

[0034] 本发明实施例还提供一种非暂态计算机可读存储介质,其上存储有计算机程序,该计算机程序被处理器执行时实现如上述任一种所述一种Web用户界面生成方法的步骤。

[0035] 本发明实施例提供的一种Web用户界面生成方法及装置,通过为响应类绑定模板,动态生成前端页面,实现事件与数据的绑定,后端提供业务响应,前端动态局部更新,提高了开发效率,降低了开发和维护的成本。

附图说明

[0036] 为了更清楚地说明本发明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一简单地介绍,显而易见地,下面描述中的附图是本发明的一些实施例,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。

- [0037] 图1是本发明实施例提供的一种Web用户界面生成方法的流程示意图；
- [0038] 图2是本发明实施例提供的一种Web用户界面生成方法的另一种流程示意图；
- [0039] 图3是本发明实施例提供的另一种Web用户界面生成方法的流程示意图；
- [0040] 图4是本发明实施例提供的一种Web用户界面生成装置的结构示意图；
- [0041] 图5是本发明实施例提供的另一种Web用户界面生成装置的结构示意图；
- [0042] 图6是根据本发明实施例提供的一种Web用户界面生成方法获取的Web用户界面示意图；
- [0043] 图7是本发明实施例提供的一种电子设备的结构示意图。

具体实施方式

[0044] 为使本发明实施例的目的、技术方案和优点更加清楚，下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0045] 在工业软件界面中，界面内含有大量的实时数据与复杂的业务处理逻辑，采用常规的Web页面中很难实现且性能低下（前端页面通常采用的是JS脚本语言），因此为了满足工业软件界面交互的高性能与业务功能的复杂性，设计出一种具备高交互复杂业务的Web页面生成方法。

[0046] 通过可视化页面建模技术，实现对界面布局、数据绑定、事件监听进行定义，由高级变成语言程序进行加载解析，动态生成可交互前端界面，界面不含有任何业务处理，仅进行用户操作监听（界面点击、数据编辑、快捷键等），转发，由后端程序进行事件响应，进而刷新页面数据或产生新的页面。

[0047] 现行的方式通常采用前后端分离进行开发，前端编写每个页面的布局、数据绑定、业务处理等编码，后端提供各种数据请求接口。此种方式需要为每个页面编写其布局样式，页面布局偏静态，很难根据业务需要进行变形操作；绑定的数据量过大会降低页面响应性能；由于是弱语言，很难实现数据的高效计算与操作；业务处理功能代码与界面布局、数据绑定代码混淆在一起，难于维护扩展及复用，带来了较高的开发与维护成本。

[0048] 本发明的目的是提供一种高交互性的动态页面生成与响应技术，用于工业软件大量数据的交互计算分析与展示，降低开发成本，提高开发效率。

[0049] 图1是本发明实施例提供的一种Web用户界面生成方法的流程示意图，如图1所示，该流程具体可以包括：

[0050] 步骤101、创建用户界面的响应类，并为所述响应类绑定预设的模板；所述模板由预设的控件组成，用于定义所述用户界面的布局；

[0051] 具体地，首先创建用户界面的响应类，然后为响应类绑定预设的模板，通常可以采用通过Java类的注解、或构造方法的传入参数完成响应类的模板的绑定。

[0052] 通过为响应类绑定预设的模板，为实现动态生成前端页面，完成事件与数据的绑定，实现后端提供业务响应，前端动态局部更新，提供了前提条件。

[0053] 步骤102、为所述模板中的控件定义事件响应函数，并实例化所述用户界面的响应类，获取用户界面的响应类实例；

[0054] 具体地,为所述模板中的控件定义事件响应函数,通常是通过 Java类的注解进行绑定;并实例化所述用户界面的响应类,获取用户界面的响应类实例;实例化所述用户界面的响应类通常需要指定父容器,作为其管理与显示位置的缺省。

[0055] 通过对模板中的控件定义事件响应函数,实例化用户界面的响应类,为实现动态生成前端页面,完成事件与数据的绑定,实现后端提供业务响应,前端动态局部更新,提供了前提条件。

[0056] 步骤103、将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;

[0057] 具体地,根据该页面UI绑定的控件实例(具备层级关系),获取该控件及其子孙控件的HTML串信息(含),并模板中各控件的超文本标记语言串信息(HTML串信息)发送至前端页面,前端接收HTML串信息,接收处理其中JS信息(用于动态布局与事件绑定),将HTML串显示在相应位置。

[0058] 通过根据UI模板定义,动态生成前端页面与前端页面的事件自动路由响应,利用编译语言的高效的数据处理效率,提示界面响应速度。

[0059] 步骤104、根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。

[0060] 具体地,用户操作界面控件(单击、双击、右键菜单等),JS捕获其控件的事件操作,将该操作信息与控件ID发送到后台服务;后端(可为本地代理)接收到前端事件信息进行处理,根据控件ID获取其对应控件实例及绑定的页面UI响应类实例,并调用该UI类绑定的事件接收函数(传入控件实例与事件信息);页面UI响应类的事件函数接收并处理,根据不同的控件的不同事件及事件参数进行相应的处理(例如:树控件局部展开、网格控件显示区间数据更新、界面数据保存等操作)。处理完成后的结果发送到Web页面(例如:创建新的UI、刷新当前控件的数据、刷新相对路径的其他控件等)。

[0061] 本发明实施例提供了一种Web用户界面生成方法,通过为响应类绑定模板,动态生成前端页面,实现事件与数据的绑定,后端提供业务响应,前端动态局部更新,提高了开发效率,降低了开发和维护的成本。

[0062] 可选地,在上述各实施例的基础上,所述将所述模板中各控件的超文本标记语言串信息发送至前端,之前还包括:

[0063] 对所述模板进行解析处理,分别获取所述模板中各控件的超文本标记语言串信息。

[0064] 具体地,对模板进行解析处理,获取定义页面布局的模板(一般由窗体、容器、网格、树、按钮、编辑框等控件组成)。根据模板顺序逐个获取每个控件定义(控件类型、名称、相对路径、风格样式、宽、高、相对布局、事件、数据绑定等)。根据控件类型创建相应控件实例(例如:Java类实例),记录该控件的布局属性、事件、数据绑定信息,并与当前页面响应类实例绑定。根据控件相对路径(当前模板中相对根节点的路径),组装控件实例的父子关系,并生成唯一标识(具有结构关系的一种ID)。根据控件绑定的事件定义(模板定义或页面UI响应类定义)与数据配置(一种基于内存表格的数据结构),动态生成该控件数据HTML串信息(含JS信息)。重复上述步骤,并逐个处理每个控件定义。

[0065] 通过对模板进行解析处理,为根据控件绑定的事件定义与数据配置,动态生成该

控件数据HTML串信息,并实现后端提供业务响应,前端动态局部更新,提供了前提条件。

[0066] 可选地,在上述各实施例的基础上,所述对所述模板进行解析处理,分别获取所述模板中各控件的超文本标记语言串信息,具体包括:

[0067] 获取所述模板中各控件的控件定义;

[0068] 基于所述控件定义获取所述模板中各控件的超文本标记语言串信息。

[0069] 图2是本发明实施例提供的一种Web用户界面生成方法的另一种流程示意图,如图2所示,根据模板顺序逐个获取每个控件定义(控件类型、名称、相对路径、风格样式、宽、高、相对布局、事件、数据绑定等);根据控件类型创建相应控件实例(例如:Java类实例),记录该控件的布局属性、事件、数据绑定信息,并与当前页面响应类实例绑定。根据控件相对路径(当前模板中相对根节点的路径),组装控件实例的父子关系,并生成唯一标识(具有结构关系的一种ID)。

[0070] 根据该页面UI绑定的控件实例(具备层级关系),获取该控件及其子孙控件的HTML串信息(含),并模板中各控件的超文本标记语言串信息(HTML串信息)发送至前端页面,前端接收HTML串信息,接收处理其中JS信息(用于动态布局与事件绑定),将HTML串显示在相应位置。

[0071] 通过根据UI模板定义,动态生成前端页面与前端页面的事件自动路由响应,利用编译语言的高效的数据处理效率,提示界面响应速度。

[0072] 可选地,在上述各实施例的基础上,所述接收前端发送的事件信息,具体包括:

[0073] 根据各个控件的ID获取各个控件对应的所述控件实例和所述用户界面的响应类实例;

[0074] 通过所述用户界面的响应类实例绑定的事件接收函数接收所述事件信息。

[0075] 具体地,后端(可为本地代理)接收到前端事件信息进行处理,根据控件ID获取其对应控件实例及绑定的页面UI响应类实例,并调用该UI类绑定的事件接收函数(传入控件实例与事件信息)。页面UI响应类的事件函数接收并处理,根据不同的控件的不同事件及事件参数进行相应的处理(例如:树控件局部展开、网格控件显示区间数据更新、界面数据保存等操作)。

[0076] 通过后端接收前端事件信息并进行处理,实现后端提供业务响应,前端动态局部更新,提供了前提条件。

[0077] 可选地,在上述各实施例的基础上,所述根据所述事件响应函数对所述事件信息进行处理,具体包括:

[0078] 调用所述事件响应函数;

[0079] 根据所述事件信息中的各控件的事件类型和事件参数进行对应的处理。

[0080] 具体地,后端(可为本地代理)接收到前端事件信息进行处理,根据控件ID获取其对应控件实例及绑定的页面UI响应类实例,调用所述事件响应函数,并调用该UI类绑定的事件接收函数(传入控件实例与事件信息)。页面UI响应类的事件函数接收并处理,根据不同的控件的不同事件及事件参数进行相应的处理,根据所述事件信息中的各控件的事件类型和事件参数进行对应的处理。

[0081] 本发明实施例提供的一种Web用户界面生成方法,通过为响应类绑定模板,动态生成前端页面,实现事件与数据的绑定,后端提供业务响应,前端动态局部更新,提高了开发

效率,降低了开发和维护的成本。

[0082] 图3是本发明实施例提供的另一种Web用户界面生成方法的流程示意图,如图3所示,该流程具体可以包括:

[0083] 步骤301、接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局;

[0084] 具体地,前端接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局。

[0085] 步骤302、获取所述超文本标记语言串信息中的JS信息,并进行处理。

[0086] 具体地,前端接收HTML串信息,接收处理其中JS信息(用于动态布局与事件绑定),将HTML串显示在相应位置。后端(可为本地代理)接收到前端事件信息进行处理,根据控件ID获取其对应控件实例及绑定的页面UI响应类实例,调用所述事件响应函数,并调用该UI类绑定的事件接收函数(传入控件实例与事件信息)。页面UI 响应类的事件函数接收并处理,根据不同的控件的不同事件及事件参数进行相应的处理,根据所述事件信息中的各控件的事件类型和事件参数进行对应的处理。

[0087] 本发明实施例提供的一种Web用户界面生成方法,通过为响应类绑定模板,动态生成前端页面,实现事件与数据的绑定,后端提供业务响应,前端动态局部更新,提高了开发效率,降低了开发和维护的成本。

[0088] 图4是本发明实施例提供的一种Web用户界面生成装置的结构示意图,如图4所示,具体包括:创建模块401:用于创建用户界面的响应类,并为所述响应类绑定预设的模板;所述模板由预设的控件组成,用于定义所述用户界面的布局;实例化模块402,用于为所述模板中的控件定义事件响应函数,并实例化所述用户界面的响应类,获取用户界面的响应类实例;交互模块403,用于将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;处理模块404,用于根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。

[0089] 具体地,首先通过创建模块401创建用户界面的响应类,然后为响应类绑定预设的模板,通常可以采用通过Java类的注解、或构造方法的传入参数完成响应类的模板的绑定。通过实例化模块402为所述模板中的控件定义事件响应函数,通常是通过Java类的注解进行绑定;并实例化所述用户界面的响应类,获取用户界面的响应类实例;实例化所述用户界面的响应类通常需要指定父容器,作为其管理与显示位置的缺省。

[0090] 交互模块403用于根据该页面UI绑定的控件实例(具备层级关系),获取该控件及其子孙控件的HTML串信息(含),并模板中各控件的超文本标记语言串信息(HTML串信息)发送至前端页面,前端接收HTML串信息,接收处理其中JS信息(用于动态布局与事件绑定),将HTML串显示在相应位置。处理模块404用于后端接收到前端事件信息进行处理,根据控件ID获取其对应控件实例及绑定的页面UI响应类实例,并调用该UI类绑定的事件接收函数;页面UI响应类的事件函数接收并处理,根据不同的控件的不同事件及事件参数进行相应的处理。处理完成后的结果发送到Web页面。

[0091] 本发明实施例提供的一种Web用户界面生成装置,通过为响应类绑定模板,动态生

成前端页面,实现事件与数据的绑定,后端提供业务响应,前端动态局部更新,提高了开发效率,降低了开发和维护的成本。

[0092] 图5是本发明实施例提供的另一种Web用户界面生成装置的结构示意图,如图5所示,具体包括:

[0093] 接收模块501,用于接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局;

[0094] 处理模块502,用于获取所述超文本标记语言串信息中的JS信息,并进行处理。

[0095] 具体地,通过接收模块501接收后端发送的超文本标记语言串信息,并通过处理模块502获取超文本标记语言串信息中的JS信息,并进行处理,接收HTML串信息,接收处理其中JS信息(用于动态布局与事件绑定),将HTML串显示在相应位置。

[0096] 图6为根据本发明实施例提供的一种Web用户界面生成方法获取的Web用户界面示意图,如图6所示,本发明实施例提供的一种Web用户界面生成方法和装置,通过为响应类绑定模板,动态生成前端页面,实现事件与数据的绑定,后端提供业务响应,前端动态局部更新,提高了开发效率,降低了开发和维护的成本。

[0097] 图7示例了一种电子设备的实体结构示意图,如图7所示,该电子设备可以包括:处理器(processor)810、通信接口(Communications Interface)820、存储器(memory)830和通信总线840,其中,处理器810,通信接口820,存储器830通过通信总线840完成相互间的通信。处理器810可以调用存储器830中的逻辑指令,以执行一种Web用户界面生成方法,该方法包括:创建用户界面的响应类,并为所述响应类绑定预设的模板;所述模板由预设的控件组成,用于定义所述用户界面的布局;为所述模板中的控件定义事件响应函数,并实例化所述用户界面的响应类,获取用户界面的响应类实例;将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。

[0098] 或者包括:

[0099] 接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局;获取所述超文本标记语言串信息中的JS信息,并进行处理。

[0100] 此外,上述的存储器830中的逻辑指令可以通过软件功能单元的形式实现并作为独立的产品销售或使用,可以存储在一个计算机可读取存储介质中。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介质中,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,或者网络设备等)执行本发明各个实施例所述方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(ROM, Read- Only Memory)、随机存取存储器(RAM, Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0101] 另一方面,本发明实施例还提供一种计算机程序产品,所述计算机程序产品包括存储在非暂态计算机可读存储介质上的计算机程序,所述计算机程序包括程序指令,当所述程序指令被计算机执行时,计算机能够执行上述各方法实施例所提供的一种Web用户界

面生成方法,该方法包括:创建用户界面的响应类,并为所述响应类绑定预设的模板;所述模板由预设的控件组成,用于定义所述用户界面的布局;为所述模板中的控件定义事件响应函数,并实例化所述用户界面的响应类,获取用户界面的响应类实例;将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。

[0102] 或者包括:

[0103] 接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局;获取所述超文本标记语言串信息中的JS信息,并进行处理。

[0104] 又一方面,本发明实施例还提供一种非暂态计算机可读存储介质,其上存储有计算机程序,该计算机程序被处理器执行时实现以执行上述各实施例提供的一种Web用户界面生成方法,该方法包括:创建用户界面的响应类,并为所述响应类绑定预设的模板;所述模板由预设的控件组成,用于定义所述用户界面的布局;为所述模板中的控件定义事件响应函数,并实例化所述用户界面的响应类,获取用户界面的响应类实例;将所述模板中各控件的超文本标记语言串信息发送至前端,接收前端发送的事件信息;根据所述事件响应函数对所述事件信息进行处理,获取处理结果,将所述处理结果发送回用户界面。

[0105] 或者包括:

[0106] 接收后端发送的超文本标记语言串信息;所述超文本标记语言串信息是基于所述后端中的为响应类绑定的预设的模板获取的,所述模板由预设的控件组成,用于定义所述用户界面的布局;获取所述超文本标记语言串信息中的JS信息,并进行处理。

[0107] 以上所描述的装置实施例仅仅是示意性的,其中所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的劳动的情况下,即可以理解并实施。

[0108] 通过以上的实施方式的描述,本领域的技术人员可以清楚地了解到各实施方式可借助软件加必需的通用硬件平台的方式来实现,当然也可以通过硬件。基于这样的理解,上述技术方案本质上或者说对现有技术做出贡献的部分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计算机可读存储介质中,如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,或者网络设备)执行各个实施例或者实施例的某些部分所述的方法。

[0109] 最后应说明的是:以上实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

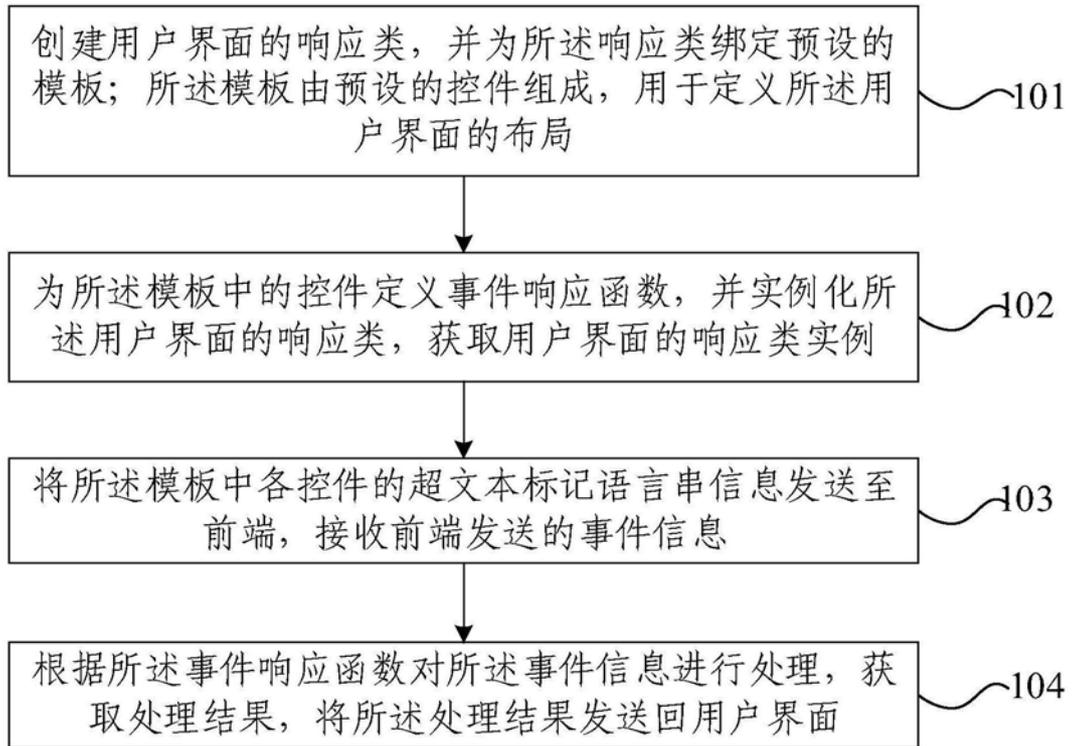


图1

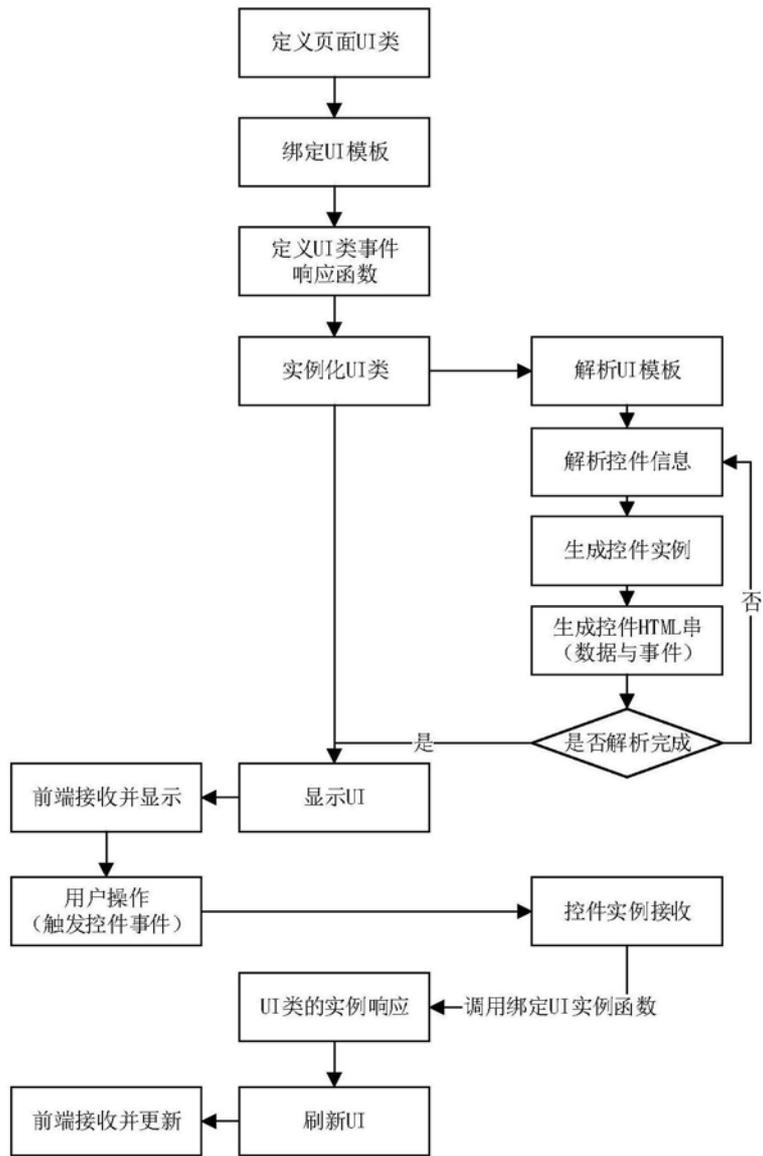


图2

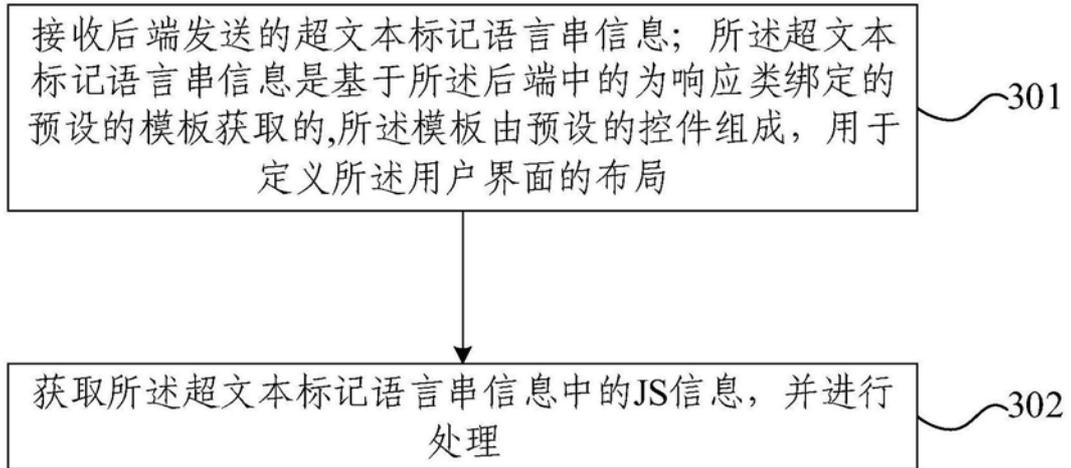


图3

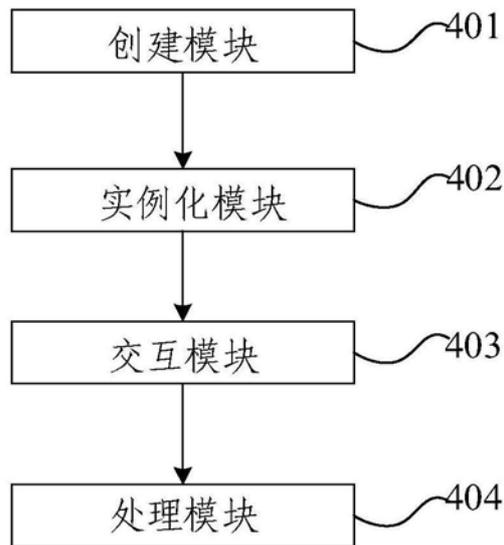


图4

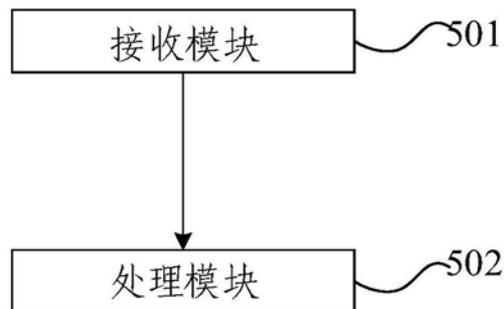


图5

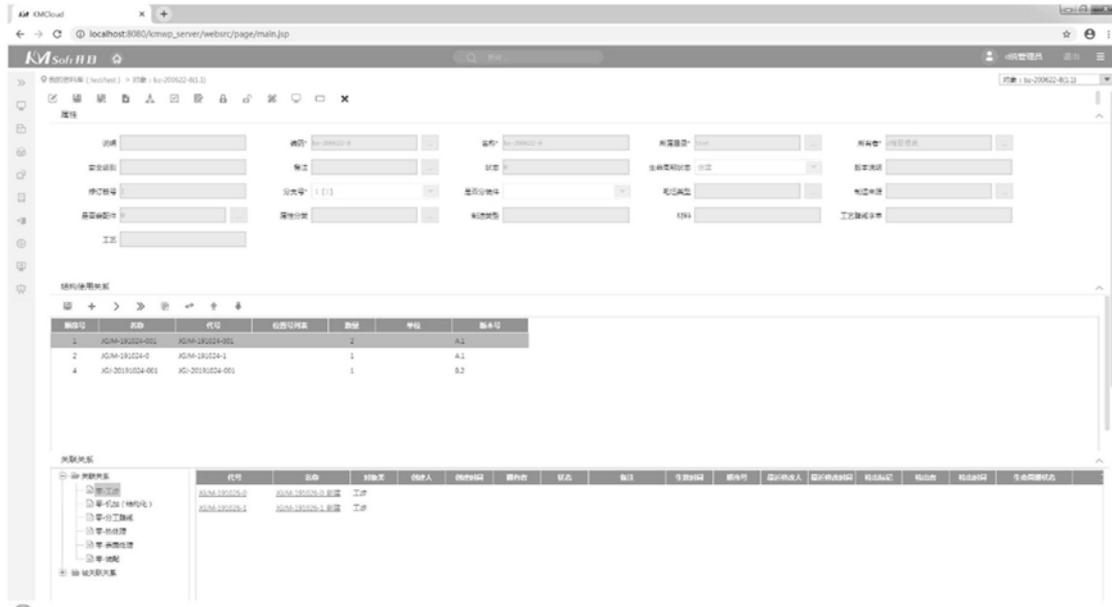


图6

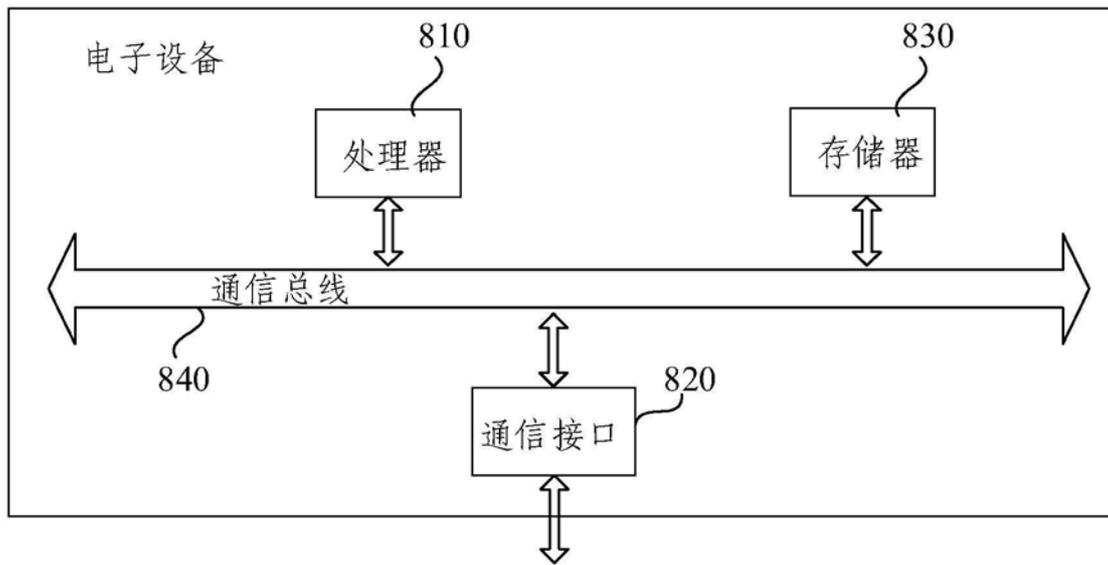


图7