



(12)发明专利申请

(10)申请公布号 CN 106528078 A

(43)申请公布日 2017. 03. 22

(21)申请号 201610847409.6

(22)申请日 2016.09.23

(71)申请人 北京五八信息技术有限公司

地址 100083 北京市海淀区学清路甲18号
中关村东升科技园学院园三层301室

(72)发明人 罗干通 彭飞 吕庆春

(74)专利代理机构 工业和信息化部电子专利中
心 11010

代理人 吴永亮

(51) Int. Cl.

G06F 9/44(2006.01)

G06Q 20/12(2012.01)

G06Q 20/22(2012.01)

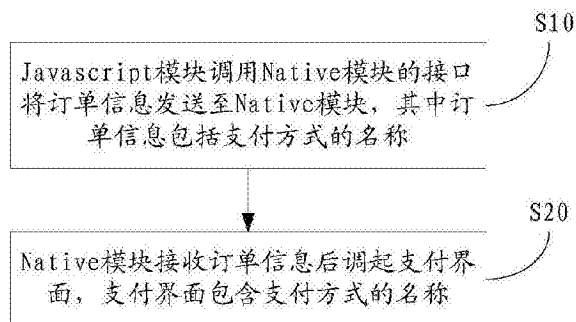
权利要求书2页 说明书6页 附图4页

(54)发明名称

一种基于react-native动态调整支付方式的方法和装置

(57)摘要

本发明提出了一种基于react-native动态调整支付方式的方法和装置,该方法,包括: Javascript模块调用Native模块的接口将订单信息发送至所述Native模块,其中所述订单信息包括支付方式的名称;所述Native模块接收所述订单信息后调起支付界面,所述支付界面包含所述支付方式的名称,及时地调整支付方式,提高用户使用移动终端App的体验。



1. 一种基于react-native动态调整支付方式的方法,其特征在于,包括:

Javascript模块调用Native模块的接口将订单信息发送至所述Native模块,其中所述订单信息包括支付方式的名称;

所述Native模块接收所述订单信息后调起支付界面,所述支付界面包含所述支付方式

的名称。

2. 根据权利要求1所述的方法,其特征在于,所述Javascript模块调用Native模块的接口将订单信息发送至所述Native模块之前,所述方法,还包括:

所述Javascript模块根据用户的操作指令确认需要支付后发送订单请求至网络服务器;

所述Javascript模块获取所述网络服务器发来的所述订单信息。

3. 根据权利要求1或2所述的方法,其特征在于,所述Native模块接收所述订单信息后调起支付界面,具体包括:

所述Native模块显示所述支付界面;

所述Native模块将所述支付界面中所述支付方式的名称链接至相应的支付服务模块;

所述Native模块初始化所述支付服务模块。

4. 根据权利要求2所述的方法,其特征在于,所述订单信息包括付款内容,所述Native模块接收所述订单信息后调起支付界面之后,所述方法,还包括:

所述Native模块获取用户从所述支付方式中选择的一种支付方式;

所述Native模块将所述付款内容发送至该种支付方式对应的支付服务模块,以使所述支付服务模块启动支付流程;

所述Native模块从所述支付服务模块获取支付状态信息后将所述支付状态信息发送至所述Javascript模块;

所述Javascript模块根据所述支付状态信息执行相应操作。

5. 根据权利要求4所述的方法,其特征在于,所述Javascript模块根据所述支付状态信息执行相应操作,具体包括:

若所述支付状态信息为支付失败并且订单未完成,所述Javascript模块将所述支付状态信息发送至所述Native模块,以使所述Native模块调起支付失败的状态界面;

若所述支付状态信息为支付成功并且订单完成,则所述Javascript模块将所述支付状态信息发送至所述Native模块,以使所述Native模块调起订单完成的状态界面;

若所述支付状态信息为支付成功并且订单未完成,则所述Native模块调起退款通知的状态界面,并且所述Javascript模块发送所述支付状态信息至所述网络服务器,以使所述网络服务器执行退款操作。

6. 一种基于react-native动态调整支付方式的装置,其特征在于,包括:

Javascript模块,用于调用Native模块的接口将订单信息发送至所述Native模块,其中所述订单信息包括支付方式的名称;

Native模块,用于接收所述订单信息后调起支付界面,所述支付界面包含所述支付方式

的名称。

7. 根据权利要求6所述的装置,其特征在于,所述Javascript模块,在调用Native模块的接口将订单信息发送至所述Native模块之前,还用于:

根据用户的操作指令确认需要支付后发送订单请求至网络服务器；
获取所述网络服务器发来的所述订单信息。

8. 根据权利要求6或7所述的装置,其特征在於,所述Native模块,包括:

显示模块,用于显示所述支付界面;

链接模块,用于将所述支付界面中所述支付方式的名称链接至相应的支付服务模块;

初始化模块,用于初始化所述支付服务模块。

9. 根据权利要求7所述的装置,其特征在於,所述订单信息包括付款内容,所述Native模块接收所述订单信息后调起支付界面之后,还用于:

获取用户从所述支付方式中选择的一种支付方式;

将所述付款内容发送至该种支付方式对应的支付服务模块,以使所述支付服务模块启动支付流程;

从所述支付服务模块获取支付状态信息后将所述支付状态信息发送至所述Javascript模块;

相应地,所述Javascript模块还用于根据所述支付状态信息执行相应操作。

10. 根据权利要求9所述的装置,其特征在於,所述Javascript模块,具体用于:

若所述支付状态信息为支付失败并且订单未完成,所述Javascript模块将所述支付状态信息发送至所述Native模块,以使所述Native模块调起支付失败的状态界面;

若所述支付状态信息为支付成功并且订单完成,则所述Javascript模块将所述支付状态信息发送至所述Native模块,以使所述Native模块调起订单完成的状态界面;

若所述支付状态信息为支付成功并且订单未完成,则所述Native模块调起退款通知的状态界面,并且所述Javascript模块发送所述支付状态信息至所述网络服务器,以使所述网络服务器执行退款操作。

11. 一种终端,其特征在於,包括根据权利要求6至10中任一项所述的装置。

一种基于react-native动态调整支付方式的方法和装置

技术领域

[0001] 本发明涉及网络支付技术领域,尤其涉及一种基于react-native动态调整支付方式的方法和装置。

背景技术

[0002] 随着网络电商App业务的发展,网络支付变得普及开来,不少非金融机构推出了支付软件开发工具包SDK,而网络电商需要接入第三方的支付SDK以方便用户支付。存在的问题是,部分支付公司并没有支付牌照或者出现支付牌照到期而导致网上在线支付无法及时完成,有时也会出现支付SDK存在bug,导致电商App崩溃,严重影响了用户使用移动终端APP的体验,并且网络电商从发现崩溃到通知第三方进行修复后重新发版需要较长的时间,这对网络电商是一个很大的损失。因此,需要一种基于react-native动态调整支付方式的方法,以解决现有技术中存在的上述技术问题。

发明内容

[0003] 本发明提供一种基于react-native动态调整支付方式的方法和装置,及时调整支付方式,提高用户使用移动终端App的体验。

[0004] 本发明采用的技术方案是:

[0005] 一种基于react-native动态调整支付方式的方法,其包括:Javascript模块调用Native模块的接口将订单信息发送至所述Native模块,其中所述订单信息包括支付方式的名称;所述Native模块接收所述订单信息后调起支付界面,所述支付界面包含所述支付方式的名称。

[0006] 优选地,所述Javascript模块调用Native模块的接口将订单信息发送至所述Native模块之前,所述方法,还包括:所述Javascript模块根据用户的操作指令确认需要支付后发送订单请求至网络服务器;所述Javascript模块获取所述网络服务器发来的所述订单信息。

[0007] 优选地,所述Native模块接收所述订单信息后调起支付界面,具体包括:所述Native模块显示所述支付界面;所述Native模块将所述支付界面中所述支付方式的名称链接至相应的支付服务模块;所述Native模块初始化所述支付服务模块。

[0008] 优选地,所述订单信息包括付款内容,所述Native模块接收所述订单信息后调起支付界面之后,所述方法,还包括:所述Native模块获取用户从所述支付方式中选择的一种支付方式;所述Native模块将所述付款内容发送至该种支付方式对应的支付服务模块,以使所述支付服务模块启动支付流程;所述Native模块从所述支付服务模块获取支付状态信息后将所述支付状态信息发送至所述Javascript模块;所述Javascript模块根据所述支付状态信息执行相应操作。

[0009] 优选地,所述Javascript模块根据所述支付状态信息执行相应操作,具体包括:若所述支付状态信息为支付失败并且订单未完成,所述Javascript模块将所述支付状态信息

发送至所述Native模块,以使所述Native模块调起支付失败的状态界面;若所述支付状态信息为支付成功并且订单完成,则所述Javascript模块将所述支付状态信息发送至所述Native模块,以使所述Native模块调起订单完成的状态界面;若所述支付状态信息为支付成功并且订单未完成,则所述Native模块调起退款通知的状态界面,并且所述Javascript模块发送所述支付状态信息至所述网络服务器,以使所述网络服务器执行退款操作。

[0010] 本发明还提供了一种基于react-native动态调整支付方式的装置,其包括: Javascript模块,用于调用Native模块的接口将订单信息发送至所述Native模块,其中所述订单信息包括支付方式的名称;Native模块,用于接收所述订单信息后调起支付界面,所述支付界面包含所述支付方式的名称。

[0011] 优选地,所述Javascript模块,在调用Native模块的接口将订单信息发送至所述Native模块之前,还用于:根据用户的操作指令确认需要支付后发送订单请求至网络服务器;获取所述网络服务器发来的所述订单信息。

[0012] 优选地,所述Native模块,包括:显示模块,用于显示所述支付界面;链接模块,用于将所述支付界面中所述支付方式的名称链接至相应的支付服务模块;初始化模块,用于初始化所述支付服务模块。

[0013] 优选地,所述订单信息包括付款内容,所述Native模块接收所述订单信息后调起支付界面之后,还用于:获取用户从所述支付方式中选择的一种支付方式;将所述付款内容发送至该种支付方式对应的支付服务模块,以使所述支付服务模块启动支付流程;从所述支付服务模块获取支付状态信息后将所述支付状态信息发送至所述Javascript模块;相应地,所述Javascript模块还用于根据所述支付状态信息执行相应操作。

[0014] 优选地,所述Javascript模块,具体用于:若所述支付状态信息为支付失败并且订单未完成,所述Javascript模块将所述支付状态信息发送至所述Native模块,以使所述Native模块调起支付失败的状态界面;若所述支付状态信息为支付成功并且订单完成,则所述Javascript模块将所述支付状态信息发送至所述Native模块,以使所述Native模块调起订单完成的状态界面;若所述支付状态信息为支付成功并且订单未完成,则所述Native模块调起退款通知的状态界面,并且所述Javascript模块发送所述支付状态信息至所述网络服务器,以使所述网络服务器执行退款操作。

[0015] 此外,本发明还提供了一种终端,其包括根据上文所述的装置。

[0016] 采用上述技术方案,本发明至少具有下列效果:

[0017] 采用本发明的基于react-native动态调整支付方式的方法,通过react-native实现了支付业务操作与支付服务操作的分离,将支付业务代码动态化,来调整支付方式,避免由发版操作存在的等待时间长的缺陷。

附图说明

[0018] 图1为本发明第一实施例的基于react-native动态调整支付方式的方法的流程图;

[0019] 图2为本发明第二实施例的基于react-native动态调整支付方式的方法的流程图;

[0020] 图3为本发明第三实施例的基于react-native动态调整支付方式的方法的流程图;

图；

[0021] 图4为本发明第四实施例的基于react-native动态调整支付方式的方法的流程图；

图；

[0022] 图5为本发明第五实施例的基于react-native动态调整支付方式的装置的方框图；

图；

[0023] 图6为本发明第六实施例的基于react-native动态调整支付方式的装置的方框图；

[0024] 图7为正常的支付界面的示意图；

[0025] 图8为采用本发明第一实施例的基于react-native动态调整支付方式的方法之后显示的支付界面的示意图。

具体实施方式

[0026] 为更进一步阐述本发明为达成预定目的所采取的技术手段及功效,以下结合附图及较佳实施例,对本发明进行详细说明如后。

[0027] 本发明提供的基于react-native动态调整支付方式的方法,可以有效地实现支付业务操作和支付服务操作的分离,及时地调整支付方式,提高用户使用移动终端App的体验。下面将详细地描述本发明的动态调整支付方式的方法及其各个步骤。

[0028] 第一实施例

[0029] 如图1所示,本实施例公开的基于react-native动态调整支付方式的方法,包括:步骤S10:Javascript模块调用Native模块的接口将订单信息发送至Native模块,其中订单信息包括支付方式的名称。以及,步骤S20:Native模块接收订单信息后调起支付界面,支付界面包含支付方式的名称。

[0030] 以上技术方案使用react-native可将支付服务操作和支付业务操作分离,由Javascript模块执行支付业务操作,获取订单信息,由此可以实现支付业务操作动态可配置,尤其是当出现支付服务模块出现崩溃的情况下,Javascript模块获取及时修改的订单信息,该订单信息包括目前可以使用的支付方式的名称,订单信息可以采用代码化的方式发送至Native模块,Native模块接收订单信息,执行支付服务操作,调起支付界面。由此在支付服务模块更新功能或者支付服务模块的支付功能出现异常时将其隐藏,保证移动终端的原生UI组件实现优秀的交互体验,避免移动终端App需要发版上线新功能存在的一系列问题。

[0031] 在该实施例中,Native模块和Javascript模块通过协商好的协议进行交互,包括代码解析协议和调用协议,可以采用现有的代码解析协议和调用协议。首先Javascript模块根据用户的输入指令判断是否需要支付,如果需要支付则由Javascript模块调用Native模块的接口时将订单信息发送给Native端,Native模块到代码化的订单信息后进行代码解析,根据订单信息判断是否调起支付界面,在调起支付界面,并且根据订单信息中的支付方式选择哪些支付可以使用,在用户支付之后,Native模块将支付状态信息发送给Javascript模块,由Javascript模块执行相应的操作。

[0032] 采用react-native的数据接口协议和格式规范如下:

[0033] callPay (options:Object,callback:Function)

[0034] 其中Object为初始化参数,其属性必须包含以下项(以下属性不做特殊说明均为必填)

[0035]

字段	类型	含义
payAction	字典	支付相关的内容,透传给支付服务

[0036]

services	字符串数组	支付服务的字符串类型的数组,数组中的内容为可提供的用户使用的支付功能,比如QQ,支付宝,微信支付等
resultCallback	回调函数	前端传递的回调函数,在完成整个支付操作,不管是成功还是失败,都会执行这个函数来告诉前端结果

[0037] 比如,以付费发帖为例,用户进行发帖时,Javascript模块判断出需要支付,则配置支付参数,然后构造调用参数,调起native模块的支付服务操作,this.finishPay即为完成支付后回调:

```
let param={
  "services": [
    "alipay",
    "weixinpay",
    "qqpay",
    "unionpay"
  ],
  "payAction": {
    "orderid": "8486856910474856970",
    "postid": "3794727185967743",
    "token": "D8he7HK74hdy7HHY8dhs7JJK"
  }
};
NativeModules.WBPay.callPay(param, this.finishPay);
```

[0038]

[0039] Native模块在接收到调用请求后唤起支付界面并根据订单信息调整支付界面的支付方式,如图7所示是一个正常的支付流程,但是如果微信支付突然不能用或者产生崩溃影响用户使用,则Javascript模块只需将订单信息中services中的weixinpay去掉即可,如图8所示。

[0040] 第二实施例

[0041] 如图2所示,在第一实施例的基础上,本实施例在步骤S10:Javascript模块调用Native模块的接口将订单信息发送至Native模块之前,动态调整支付方式的方法,还包括:步骤S30:Javascript模块根据用户的操作指令确认需要支付后发送订单请求至网络服务器;步骤S40:Javascript模块获取网络服务器发来的订单信息。通过网络服务器发送修改后的订单信息来实现动态调整,因为网络服务器具有监控支付服务模块是否正常运行功能。

[0042] 第三实施例

[0043] 如图3所示,在第一实施例的基础上,本实施例的步骤S20:Native模块接收订单信息后调起支付界面,具体包括:步骤S200:Native模块显示支付界面;步骤S201:Native模块

将支付界面中支付方式的名称链接至相应的支付服务模块；步骤S202:Native模块初始化支付服务模块。由此,用户可以操作支付界面上显示的支付方式的名称进行支付操作。

[0044] 第四实施例

[0045] 如图4所示,本实施例提供的动态调整支付方式的方法中,订单信息包括付款内容,付款内容包括付款款项、终端IP地址和收款方等信息。步骤S20:Native模块接收订单信息后调起支付界面之后,本实施例的动态调整支付方式的方法,还包括:步骤S50:Native模块获取用户从支付方式中选择的一种支付方式;步骤S60:Native模块将付款内容发送至该种支付方式对应的支付服务模块,以使支付服务模块启动支付流程;步骤S70:Native模块从支付服务模块获取支付状态信息后将支付状态信息发送至Javascript模块;步骤S80:Javascript模块根据支付状态信息执行相应操作。

[0046] 作为优选地,步骤S80:Javascript模块根据支付状态信息执行相应操作,具体包括:若支付状态信息为支付失败并且订单未完成,Javascript模块将支付状态信息发送至Native模块,以使Native模块调起支付失败的状态界面;若支付状态信息为支付成功并且订单完成,则Javascript模块将支付状态信息发送至Native模块,以使Native模块调起订单完成的状态界面;若支付状态信息为支付成功并且订单未完成,则Javascript模块将支付状态信息发送至Native模块,以使Native模块调起退款通知的状态界面,并且Javascript模块发送支付状态信息至网络服务器,以使网络服务器执行退款操作,可以参考现有技术中网络服务器执行退款操作的方法。

[0047] 下面以网站发帖需要付费来阐述本发明的基于react-native动态调整支付方式的方法的执行流程:

[0048] 1.用户在发帖界面进行发帖,帖子完成后需要发布时,Javascript模块判断是否需要支付,若需要支付,则执行2,否则跳到8;

[0049] 2.Javascript模块向网络服务器发起订单请求,以获取订单信息,该订单信息包括支付方式的名称;

[0050] 3.Javascript模块将订单信息封装后调起native模块的接口;

[0051] 4.native模块接收封装后的订单信息后解析,确定可以使用的支付方式;

[0052] 5.native模块初始化可以使用的支付方式对应的支付服务模块,然后将订单信息发送到上述支付服务模块;

[0053] 6.在用户支付后,native模块获取订单的支付状态信息,通过回调将订单状态信息发送至Javascript模块;

[0054] 7.Javascript模块根据支付状态信息进行相应操作,若支付成功并且发帖完成,则提示用户发帖成功;若支付成功是发帖失败,则Javascript模块将支付状态信息发送至Native模块,以使Native模块调起退款通知的状态界面,并且Javascript模块将订单状态信息发送至网络服务器,由网络服务器执行退款操作;若支付状态信息为支付失败但是订单失败,Javascript模块将支付状态信息发送至Native模块,以使Native模块调起支付失败的状态界面。

[0055] 8.不需要支付则直接发布。

[0056] 第五实施例

[0057] 如图5所示,本实施例的基于react-native动态调整支付方式的装置,包括:

Javascript模块10,用于调用Native模块的接口将订单信息发送至Native模块,其中订单信息包括支付方式的名称;Native模块20,用于接收订单信息后调起支付界面,支付界面包含支付方式的名称。

[0058] 作为优选地,Javascript模块10,在调用Native模块20的接口将订单信息发送至Native模块10之前,还用于:根据用户的操作指令确认需要支付后发送订单请求至网络服务器;获取网络服务器发来的订单信息。

[0059] 另外,订单信息包括付款内容,Native模块20接收订单信息后调起支付界面之后,还用于:获取用户从支付方式中选择的一种支付方式;将付款内容发送至该种支付方式对应的支付服务模块,以使支付服务模块启动支付流程;从支付服务模块获取支付状态信息后将支付状态信息发送至Javascript模块10;相应地,Javascript模块10还用于根据支付状态信息执行相应操作。

[0060] 此外,Javascript模块10,具体用于:若支付状态信息为支付失败并且订单未完成,Javascript模块将支付状态信息发送至Native模块,以使Native模块调起支付失败的状态界面;若支付状态信息为支付成功并且订单完成,则Javascript模块将支付状态信息发送至Native模块,以使Native模块调起订单完成的状态界面;若支付状态信息为支付成功并且订单未完成,则Native模块调起退款通知的状态界面,并且Javascript模块发送支付状态信息至网络服务器,以使网络服务器执行退款操作。

[0061] 第六实施例

[0062] 如图6所示,本实施例的动态调整支付方式的装置中,Native模块20,包括:显示模块200,用于显示支付界面;链接模块201,用于将支付界面中支付方式的名称链接至相应的支付服务模块;初始化模块202,用于初始化支付服务模块。

[0063] 第七实施例

[0064] 此外,本发明还提供了一种终端,其包括根据上文所述的装置。

[0065] 通过具体实施方式的说明,应当可对本发明为达成预定目的所采取的技术手段及功效得以更加深入且具体的了解,然而所附图示仅是提供参考与说明之用,并非用来对本发明加以限制。

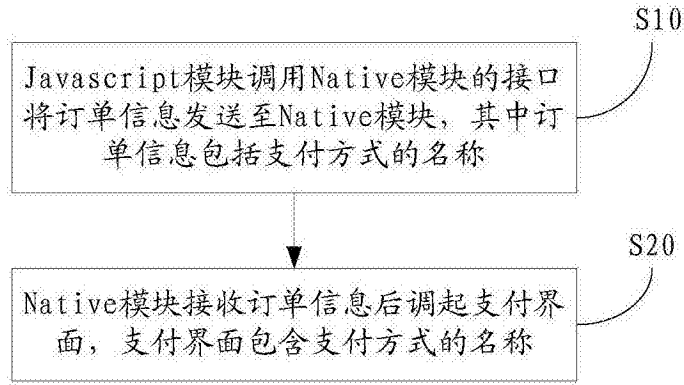


图1

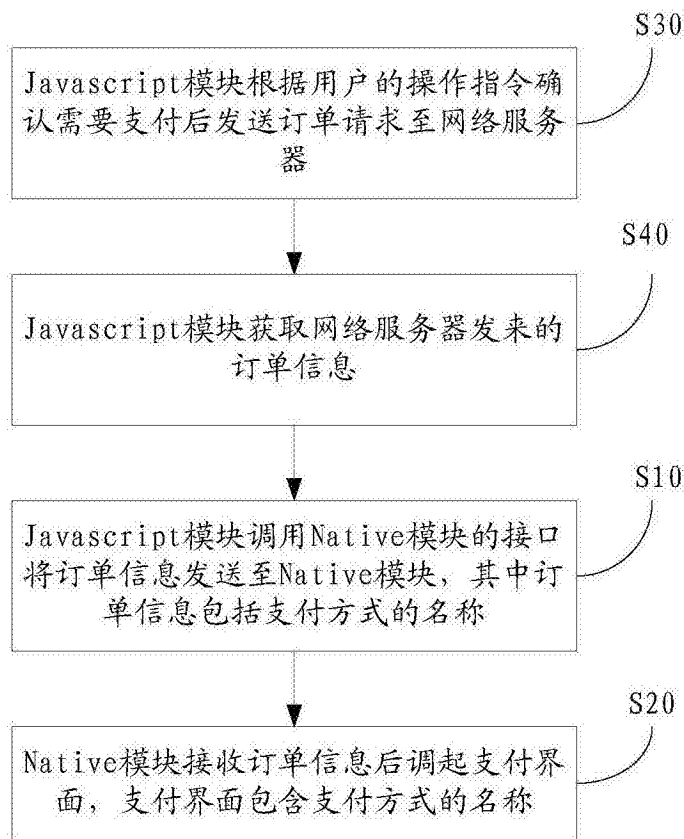


图2

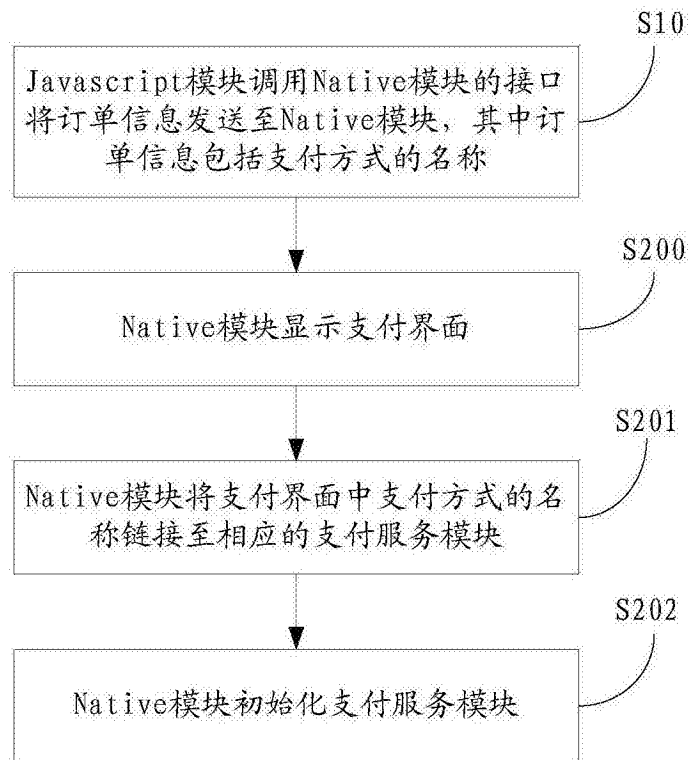


图3

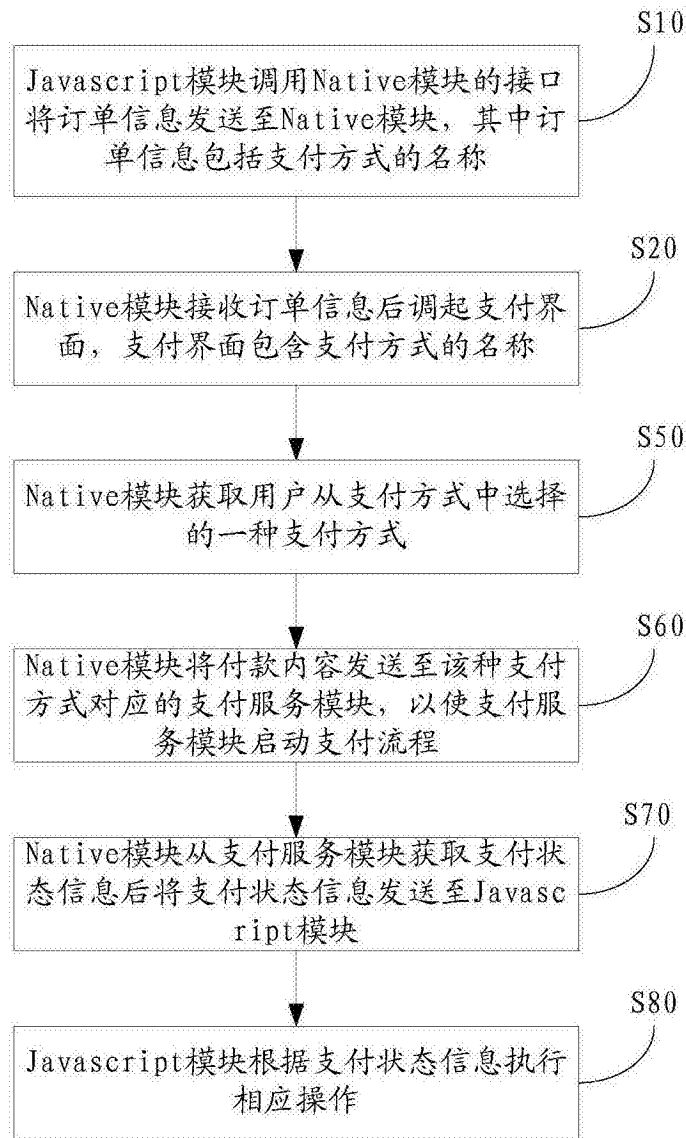


图4

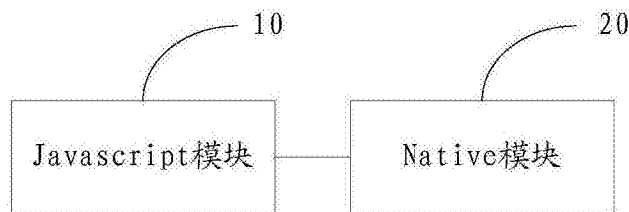


图5

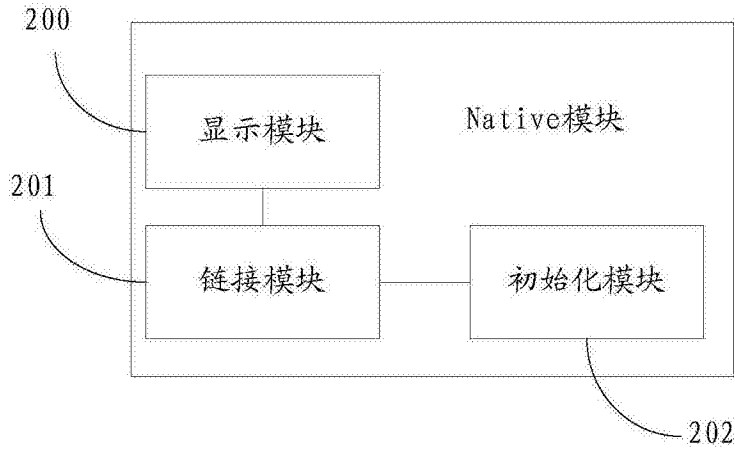


图6



图7



图8