

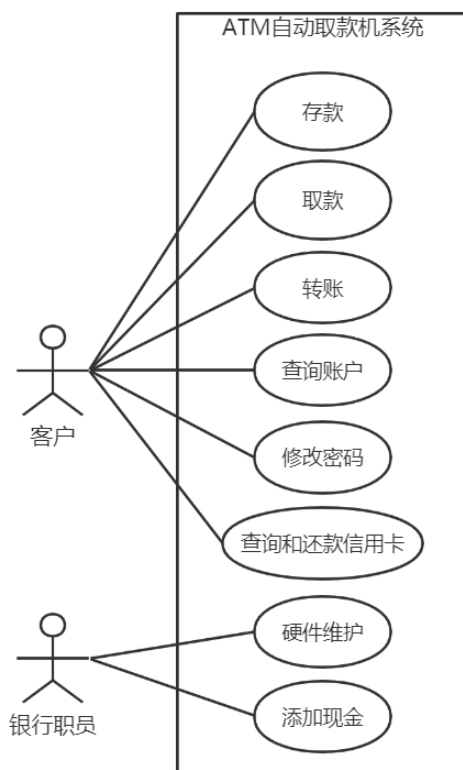
软件工程导论作业参考作答

作业 1

第 1 题：

1. ATM 自动取款机是银行在银行营业大厅、超市、商业机构、机场、车站、码头和闹市区设置的一种小型机器，利用一张信用卡大小的胶卡上的磁带〔或芯片卡上的芯片〕记录客户的基本户口资料，让客户可以透过机器进行提款、存款、转帐等银行柜台服务。
 - a) 客户将银行卡插入读卡器，读卡器识别卡的真伪，并在显示器上提示输入密码。
 - b) 客户通过键盘输入密码，取款机验证密码是否有效。如果密码错误提示错误信息，如果正确，提示客户进行选择操作的业务。
 - c) 客户根据自己的需要可进行银行账号的存款、取款、查询账户、转账、修改密码的操作。
 - d) 客户还可以进行查询和还款自己的信用卡账号的操作。
 - e) 在客户选择后显示器进行交互提示和操作确认等信息。
 - f) 操作完毕后，客户可自由选择打印或不打印凭条。
 - g) 银行职员可进行对 ATM 自动取款机的硬件维护和添加现金的操作。请对上述需求分析画出用例图和用例说明（用例表）。

参与者：客户、银行职员



可以增加登录系统、打印凭条等用例，并考虑用例之间的关系，也可以将它们作为流程

场景中的几个流程。

括号里的名称是对应教科书上用例说明模板里的名称，类似即可。

用例编号	1
用例名称（用例）	存款
参与者	客户
用例描述（情境目标）	客户使用银行卡在 ATM 机上进行存款操作
前置条件（前提条件）	银行卡插卡、密码验证有效，成功登录系统
触发条件（触发器）	客户在 ATM 机上点击存款按钮
后置条件	客户期望金额成功存入，存款信息同步到数据库
正常流程（场景）	1、客户将银行卡插入读卡器并输入密码登录系统。 2、客户点击存款按钮，跳转存款界面。 3、客户将钱投入 ATM 机中。 4、ATM 机校验存款数量，将无效人民币退回，并将有效存款金额反馈给客户。 5、客户点击确认存款金额。 6、ATM 机将存款信息同步数据库，并提示客户存款成功。 7、客户选择是否打印凭条。
异常流程（异常处理）	1、读卡器未识别并退卡异常：银行卡为假或该 ATM 机不支持该发行行的操作，客户可以到人工柜台请求帮助或更换银行卡。 2、密码输入错误：客户重新输入密码，超过 3 次吞卡，需联系银行职员进行取卡解锁等操作。 3、存在未识别的钞票：ATM 机退钞，客户取出后可重新投入。 4、ATM 显示存款金额错误：联系银行职员进行修复。

用例编号	2
用例名称（用例）	取款
参与者	客户
用例描述（情境目标）	客户使用银行卡在 ATM 机上进行取款操作
前置条件（前提条件）	银行卡插卡、密码验证有效，成功登录系统
触发条件（触发器）	客户在 ATM 机上点击取款按钮
后置条件	客户期望金额成功取出，取款信息同步到数据库
正常流程（场景）	1、客户将银行卡插入读卡器并输入密码登录系统。 2、客户点击取款按钮，跳转取款界面。 3、客户选择或输入取款金额。 4、ATM 机校验当前账户余额是否大于取款金额，大于则将对应数量人民币返回客户，并同步数据库。小于则提示用户，跳

	转 3。 5、客户从出钞口取出现金，选择是否打印凭条。
异常流程（异常处理）	1、读卡器未识别卡并退卡：银行卡为假或该 ATM 机不支持该发行行的操作，客户可以到人工柜台请求帮助或更换银行卡。 2、密码输入错误：客户重新输入密码，超过 3 次吞卡，需联系银行职员进行取卡解锁等操作。 3、取款金额超额：重新输入取款金额。 4、当前银行卡被冻结：显示器提示取款失败。

用例编号	3
用例名称（用例）	转账
参与者	客户
用例描述（情境目标）	客户使用银行卡在 ATM 机上进行转账操作
前置条件（前提条件）	银行卡插卡、密码验证有效，成功登录系统
触发条件（触发器）	客户在 ATM 机上点击转账按钮
后置条件	客户期望转账金额被转账到指定账户，转账信息同步到数据库
正常流程（场景）	1、客户将银行卡插入读卡器并输入密码登录系统。 2、客户点击转账按钮，跳转转账界面。 3、客户输入转账金额和转入账户。 4、ATM 机系统校验账户是否有效，余额是否大于取款金额，均有效则将数据同步数据库。否则提示用户，跳转 3。 5、客户转账成功后选择是否打印凭条。
异常流程（异常处理）	1、读卡器未识别卡并退卡：银行卡为假或该 ATM 机不支持该发行行的操作，客户可以到人工柜台请求帮助或更换银行卡。 2、密码输入错误：客户重新输入密码，超过 3 次吞卡，需联系银行职员进行取卡解锁等操作。 3、ATM 机发现对方账户不存在：提示转账失败，用户目标账户不存在，请用户重新输入转账账户或者退出该项服务。 4、ATM 机发现卡内余额不足：提示转账失败，余额不足，请用户重新输入转账金额或者退出该项服务。 5、用户输入不合法：提示转账失败，请用户重新输入信息。 6、当前银行卡被冻结：显示器提示转账失败，无法转账。

用例编号	4
用例名称（用例）	查询账户
参与者	客户
用例描述（情境目标）	客户使用银行卡在 ATM 机上进行查询账户操作
前置条件（前提条件）	银行卡插卡、密码验证有效，成功登录系统

件)	
触发条件（触发器）	客户在 ATM 机上点击查询账户按钮
后置条件	ATM 机将客户账户信息成功显示
正常流程（场景）	1、客户将银行卡插入读卡器并输入密码登录系统。 2、客户点击查询账户按钮，跳转查询账户界面。 3、客户选择要查询信息。 4、ATM 机系统将客户查询信息返回在显示器上。
异常流程（异常处理）	1、读卡器未识别并退卡异常：银行卡为假或该 ATM 机不支持该发行行的操作，客户可以到人工柜台请求帮助或更换银行卡。 2、密码输入错误：客户重新输入密码，超过 3 次吞卡，需联系银行职员进行取卡解锁等操作。

用例编号	5
用例名称（用例）	修改密码
参与者	客户
用例描述（情境目标）	客户使用银行卡在 ATM 机上进行修改密码操作
前置条件（前提条件）	银行卡插卡、密码验证有效，成功登录系统
触发条件（触发器）	客户在 ATM 机上点击修改密码按钮
后置条件	ATM 机系统将客户修改后的密码同步数据库。
正常流程（场景）	1、客户将银行卡插入读卡器并输入密码登录系统。 2、客户点击修改密码按钮，跳转修改密码界面。 3、客户输入新密码并再次确认输入。 4、ATM 机校验修改后的密码有效，并同步数据库，提示用户修改成功。
异常流程（异常处理）	1、读卡器未识别并退卡异常：银行卡为假或该 ATM 机不支持该发行行的操作，客户可以到人工柜台请求帮助或更换银行卡。 2、密码输入错误：客户重新输入密码，超过 3 次吞卡，需联系银行职员进行取卡解锁等操作。 3、ATM 机发现用户两次输入新密码不一致：ATM 机显示两次密码不一致，请用户重新输入新密码。 4、ATM 机发现该账户处于特殊状态：提示客户操作失败。

（可以通过银行卡（借记卡）余额转账方式来还款绑定的相关信用卡，也可以使用信用卡通过插卡登录系统然后现金还款，这里写的是信用卡插卡登录系统然后现金还款的方式。）

用例编号	6
用例名称（用例）	查询和还款信用卡
参与者	客户

用例描述（情境目标）	客户使用信用卡在 ATM 机上进行查询和还款信用卡操作
前置条件（前提条件）	信用卡插卡、密码验证有效，成功登录系统
触发条件（触发器）	客户在 ATM 机上点击信用卡查询还款按钮
后置条件	ATM 机系统显示用户信用卡信息，当客户还款后，将信用卡还款信息同步数据库并显示。
正常流程（场景）	<ol style="list-style-type: none"> 1、客户将信用卡插入读卡器并输入密码登录系统。 2、客户点击信用卡查询还款按钮，跳转信用卡查询还款界面。 3、客户查询信用卡欠款信息，ATM 机进行显示。 4、客户点击或输入还款金额，并将钱投入 ATM 机中。 5、ATM 机校验存款数量，将无效人民币退回，并将有效还款金额反馈给客户。 6、客户点击确认还款金额。 7、ATM 机将还款信息同步数据库，并提示客户还款成功和剩余还款金额 8、客户选择是否打印凭条。
异常流程（异常处理）	<ol style="list-style-type: none"> 1、读卡器未识别并退卡异常：信用卡为假或该 ATM 机不支持该发行行的操作，客户可以到人工柜台请求帮助或更换卡。 2、密码输入错误：客户重新输入密码，超过 3 次吞卡，需联系银行职员进行取卡解锁等操作。 3、ATM 机发现还款现金不足：提示还款失败，现金不足，请用户重新输入还款金额或者退出该项服务。 4、ATM 机发现现金还款过多：提示还款失败，金额不对，请用户重新输入还款金额。 5、信用卡信息拉取失败：用户点击刷新。

用例编号	7
用例名称（用例）	硬件维护
参与者	银行职员
用例描述（情境目标）	银行职员对 ATM 机进行硬件维护
前置条件（前提条件）	银行职员身份校验成功
触发条件（触发器）	ATM 机到达维护周期，或者机器发生故障
后置条件	ATM 机被成功维护。
正常流程（场景）	<ol style="list-style-type: none"> 1、机器到达周期提示维护，或后台数据检测到 ATM 机无法正常服务，或客户反映机器故障后，银行职员将进行机器维护。 2、ATM 机被成功维护后继续运行。
异常流程（异常处理）	<ol style="list-style-type: none"> 1、ATM 机硬件损坏严重：银行职员上报，需要专业人员维修或更换机器

用例编号	8
用例名称（用例）	添加现金
参与者	银行职员
用例描述（情境目标）	银行职员向 ATM 机添加现金
前置条件（前提条件）	银行职员身份校验成功
触发条件（触发器）	ATM 机提示机器现金余额即将不足，银行职员选择添加现金功能。
后置条件	ATM 成功添加现金，更新当前现金余额。
正常流程（场景）	1、银行职员选择添加现金功能。 2、银行职员将钱投入 ATM 机中。 3、ATM 机校验现金数量，将无效人民币退回，并将有效现金数量反馈给银行职员。 4、银行职员点击确认添加现金数量。 5、ATM 机系统更新当前现金余额，并将余额反馈给银行职员。
异常流程（异常处理）	1、存在未识别的钞票：ATM 机退钞，取出后可重新投入。

第 2 题：

某软件安全公司需要实现一个安卓灰色应用检测系统（提供在线网站服务）。

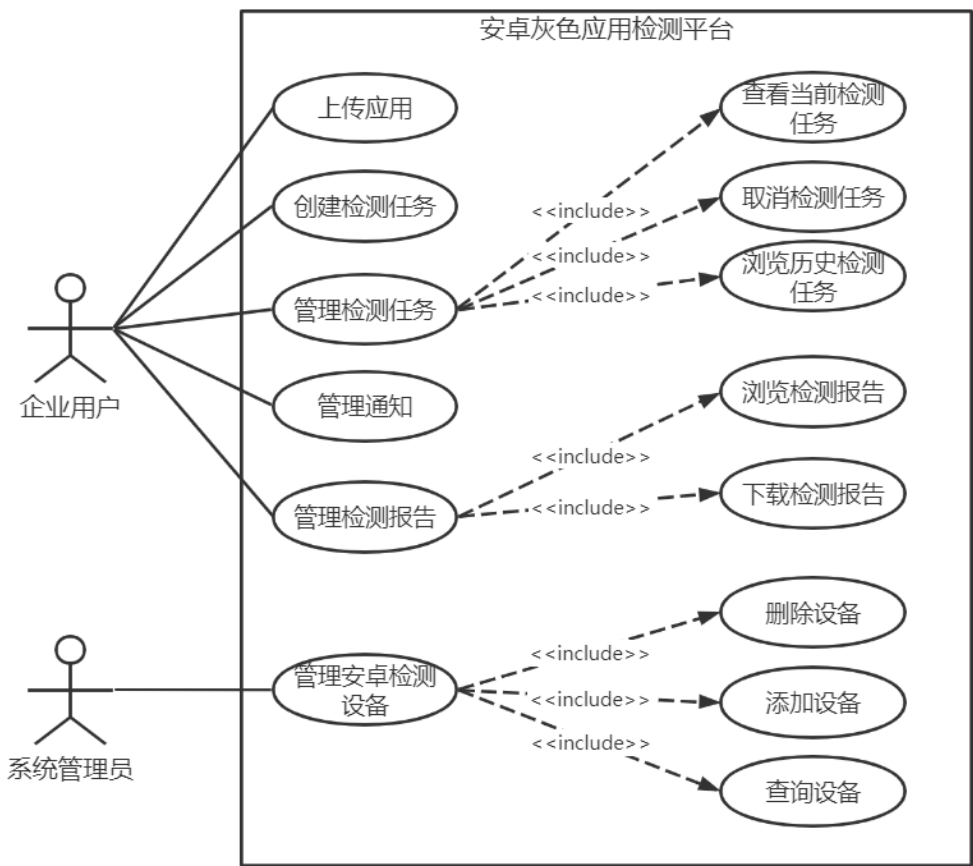
该系统的功能需求如下
 安卓灰色应用检测系统的用户主要有两类，第一类是企业用户，第二类是系统管理员。

- a) 企业用户一般是安卓应用市场上架审核人员，审核人员可以使用检测系统去审核即将在应用市场上架的新应用。对于那些通过审核、在应用市场成功上架的应用，审核人员也可以使用检测系统对其进行抽查。企业用户一次上传一个或多个应用文件到检测平台，并创建对应的检测任务，一个检测任务对应一个应用。一旦检测到灰色应用，检测系统会在 web 客户端上发出通知，并将检测过程中的关键数据组装成检测报告以供企业用户浏览和下载。检测报告中需要包含灰色应用的基本信息，比如应用名称、应用包名、版本号、图标等以及灰色应用非法更新的应用界面截图和对应地区。企业用户可以查看、取消正在进行的检测任务，也可以浏览历史检测任务的数据。
- b) 除了企业用户，检测系统还有一类用户是系统管理员。因为本文设计的动态检测技术在安卓手机上运行待检测应用，所以系统管理员需要对这些安卓检测设备进行管理。数据库中需要记录所有检测设备的状态。当检测设备老旧或者损坏，系统管理员需要更换这些设备。当检测设备的数量达不到企业用户的业务量需求时，系统管理员需要向检测系统中添加安卓检测设备。尽管 adb 工具支持安卓手机和服务器通过 TCP 协议进行无

线通信,但是无线通信的效率较低,因此系统管理员需要利用 USB 连接安卓检测设备,连接成功后,系统管理员还要对检测设备做一些初始化设置,保证和服务器连接的检测设备能够正常工作。

请对上述需求分析画出用例图和用例说明(用例表)。

参与者:企业用户、系统管理员



用例编号	1
用例名称	上传应用并创建检测任务
用例参与者	企业用户
用例描述	企业用户上传应用后可以创建检测任务
前置条件	企业用户成功登录检测平台
触发条件	企业用户点击上传应用按钮
后置条件	检测平台将企业用户创建的任务添加到任务调度列表中
正常流程	1. 企业用户点击上传应用按钮,在本地选择应用文件上传。企业用户可一次上传多个应用。企业用户点击确定,检测平台以进度条的方式给企业用户反馈上传进度。 2. 检测平台校验企业用户上传的应用文件格式是否正确。 3. 检测平台分析并在网页展示应用的基本信息,包括应用包名、版本号和应用名称。 4. 检测平台提示企业用户确认应用信息是否正确。如果不正确,用户选择重新上传应用按钮。

	5. 企业用户点击创建检测任务按钮。
异常流程	1. 如果应用包格式错误（不以.apk 结尾）或者检测平台解析应用包的基本信息失败，检测平台提示企业用户错误信息类型。 2. 创建任务用例终止，检测平台重定向到上传应用页面。

用例编号	2
用例名称	查看当前检测任务
用例参与者	企业用户
用例描述	企业用户可以在 web 页面上查看正在进行中的检测任务
前置条件	企业用户成功创建过检测任务，且检测记录还没有被用户删除
触发条件	企业用户点击任务记录
后置条件	企业用户浏览检测任务详细信息
正常流程	1. 企业用户点击任务记录。 2. 企业用户选择当前检测任务并进行查看。 3. 平台给企业用户展示检测的详细信息，包括应用的基本信息、应用最后一次的运行时间以及是否是灰色应用等。
异常流程	无

用例编号	3
用例名称	取消检测任务
参与者	企业用户
用例描述	企业用户可以随时取消一个进行中的检测任务，取消成功后，检测系统会修改数据库中该任务的相关字段，该任务不会再出现在调度列表中
前置条件	企业用户成功创建了一个检测任务且任务还在进行中
触发条件	企业用户点击取消任务按钮
后置条件	检测平台将对应的任务从检测队列中删除
正常流程	1. 企业用户选择一个正在进行中的任务 2. 企业用户点击取消任务 3. 检测平台提示企业用户确认取消检测 4. 企业用户点击确认，检测平台告知企业用户检测任务终止
异常流程	无

用例编号	4
用例名称	浏览历史检测任务
参与者	企业用户
用例描述	企业用户可以查看历史检测任务数据，如果检测记录过多，可以根据需要选择时间久远的检测记录将其删除
前置条件	检测系统中存在已经结束的检测任务，且检测记录没有被企业用户删除

触发条件	企业用户点击查看检测历史任务按钮
后置条件	企业用户浏览历史检测任务详细数据
正常流程	<ol style="list-style-type: none"> 1. 企业用户点击任务记录。 2. 企业用户选择一条历史检测记录。 3. 平台给企业用户展示检测的详细信息。 4. 企业用户选择删除检测记录。 5. 检测平台将此条检测记录从数据库中删除。
异常流程	无

用例编号	5
用例名称	管理通知
参与者	企业用户
用例描述	检测平台在检测出灰色应用或者完成一个检测任务后会给企业用户发送通知，企业用户在通知列表中可以浏览删除这些通知。
前置条件	无
触发条件	企业用户点击通知列表
后置条件	企业用户成功查看通知的具体内容或删除通知
正常流程	<ol style="list-style-type: none"> 1. 企业用户点击通知列表，选择需要查看的通知。 2. 检测平台向企业用户展示通知。 3. 若用户选择删除通知，检测平台将删除此条通知
异常流程	无

用例编号	6
用例名称	浏览检测报告
参与者	企业用户
用例描述	检测平台在检测出灰色应用后会生成一份检测报告并通过通知列表通知企业用户，企业用户可以在 web 客户端浏览检测报告。
前置条件	存在可以浏览下载的灰色应用检测报告
触发条件	企业用户选择指定的灰色应用， 点击阅读报告
后置条件	企业用户成功阅读报告， 查看灰色应用的运行信息
正常流程	<ol style="list-style-type: none"> 1. 检测平台向用户展示灰色应用信息。 2. 企业用户选择一个特定的灰色应用。 3. 用户点击查看检测报告。 4. 检测平台向企业用户展示检测报告。
异常流程	无

用例编号	7
用例名称	下载检测报告
参与者	企业用户
用例描述	检测平台在检测出灰色应用后会在通知列表中生成一个通知并

	将检测数据组装成一份检测报告，企业用户可以从检测平台下载灰色应用的检测报告，报告格式为 PDF。
前置条件	存在可以浏览下载的灰色应用检测报告
触发条件	企业用户选择指定的灰色应用，点击下载报告
后置条件	企业用户成功下载检测报告
正常流程	<ol style="list-style-type: none"> 1. 检测平台向用户展示灰色应用信息。 2. 企业用户选择一个特定的灰色应用。 3. 企业用户点击下载检测报告。 4. 检测平台提示企业用户报告下载成功。
异常流程	无

用例编号	8
用例名称	删除设备
参与者	系统管理员
用例描述	系统管理员可以在检测平台上查询所有安卓检测设备的状态，每一个检测设备拥有一个唯一的 ID 号，通过 ID 号可以区分检测设备。系统管理员确定要删除的设备 ID 后，可以在检测平台上选择对应的设备点击删除，服务器会断开和该设备的 adb 连接，并在数据库中更新该设备的相关记录，包括设备状态和停止使用时间。系统管理员可以将检测设备取走。
前置条件	某些安卓检测设备出现故障或过于老旧，不能再被检测系统使用
触发条件	系统管理员选择一个检测设备，点击删除
后置条件	系统管理员成功删除对应的安卓检测设备
正常流程	<ol style="list-style-type: none"> 1. 系统管理员用超级账号登录检测平台。 2. 系统管理员进入设备管理页面。 3. 系统管理员确定要删除设备的 ID 号。 4. 系统管理员选中对应的设备后点击删除。 5. 检测平台提示系统管理员设备删除成功。 6. 系统管理员可以将检测设备取走。
异常流程	无

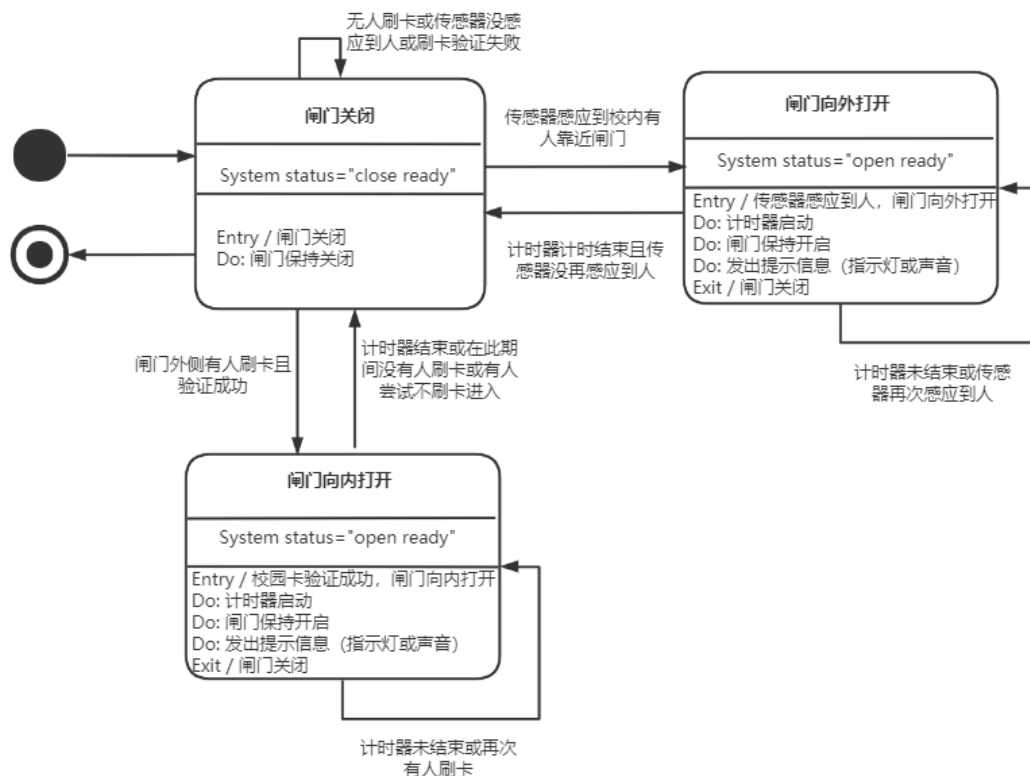
用例编号	9
用例名称	添加设备
参与者	系统管理员
用例描述	如果出现安卓检测设备不足或者设备损坏的情况，那么系统管理员需要增加安卓检测设备。检测系统会检查新设备和服务器是否成功的进行了 adb 连接。如果成功连接，检测系统会在数据库中插入记录，包括设备的 ID 号，添加时间等信息。
前置条件	检测设备数量不够或者需要更换
触发条件	系统管理员点击加入设备按钮
后置条件	新的安卓检测设备成功连接到服务器上

正常流程	<ol style="list-style-type: none"> 1. 系统管理员用超级账号登录检测平台。 2. 系统管理员进入检测设备管理页面。 3. 系统管理员点击添加设备。 4. 系统管理员用 USB 数据线连接检测设备和服务器。 5. 系统管理员输入设备的 ID 号, 检测系统会提示是否成功的进行了 adb 连接。 6. 如果检测设备和服务器成功的进行了 adb 连接, 检测系统会更新数据库。 7. 系统管理员对检测设备做一些初始化设置, 保证和服务器连接的检测设备能够正常工作。
异常流程	无

作业 2

第 1 题:

请仔细观察学校校门进出口闸机的工作方式, 绘制单个闸机工作状态图。并结合状态图, 说明学校门口闸机与其他闸机 (例如高铁进站口闸机、图书馆进口闸机) 在功能的主要区别。

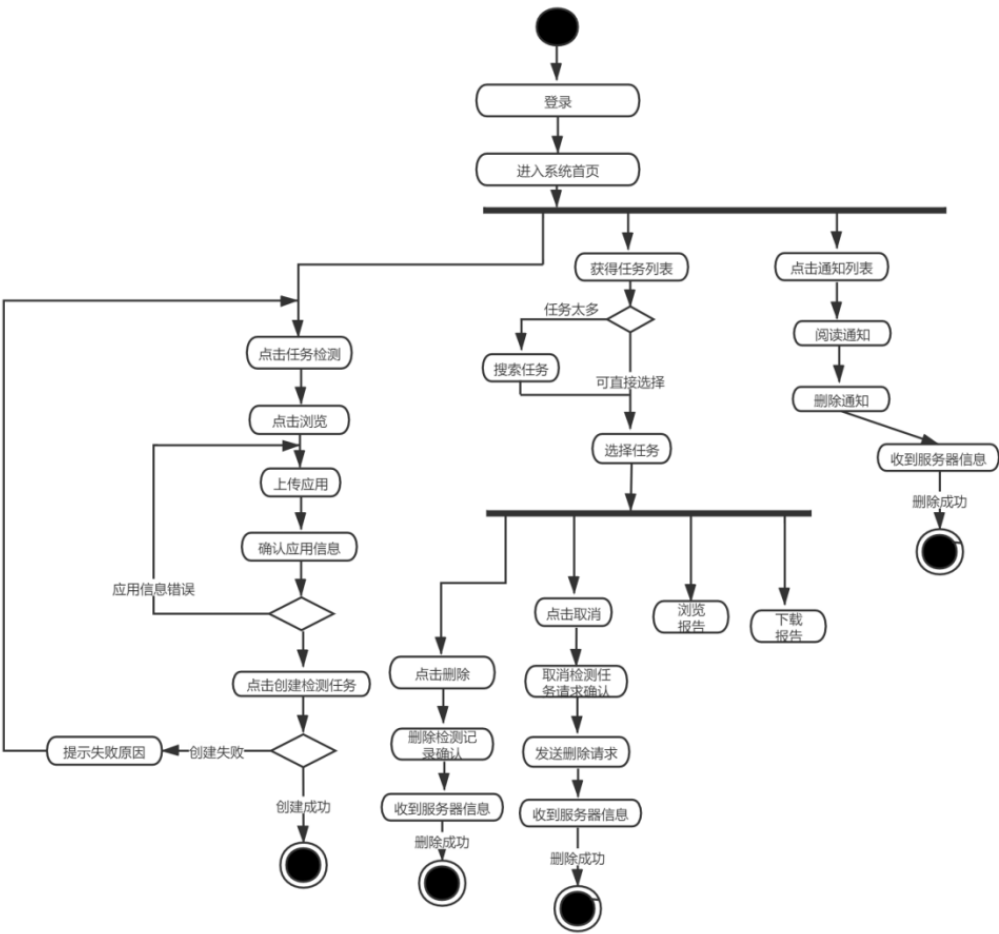


学校门口闸机与其他闸机(例如高铁进站口闸机、图书馆进口闸机)在功能的主要区别:

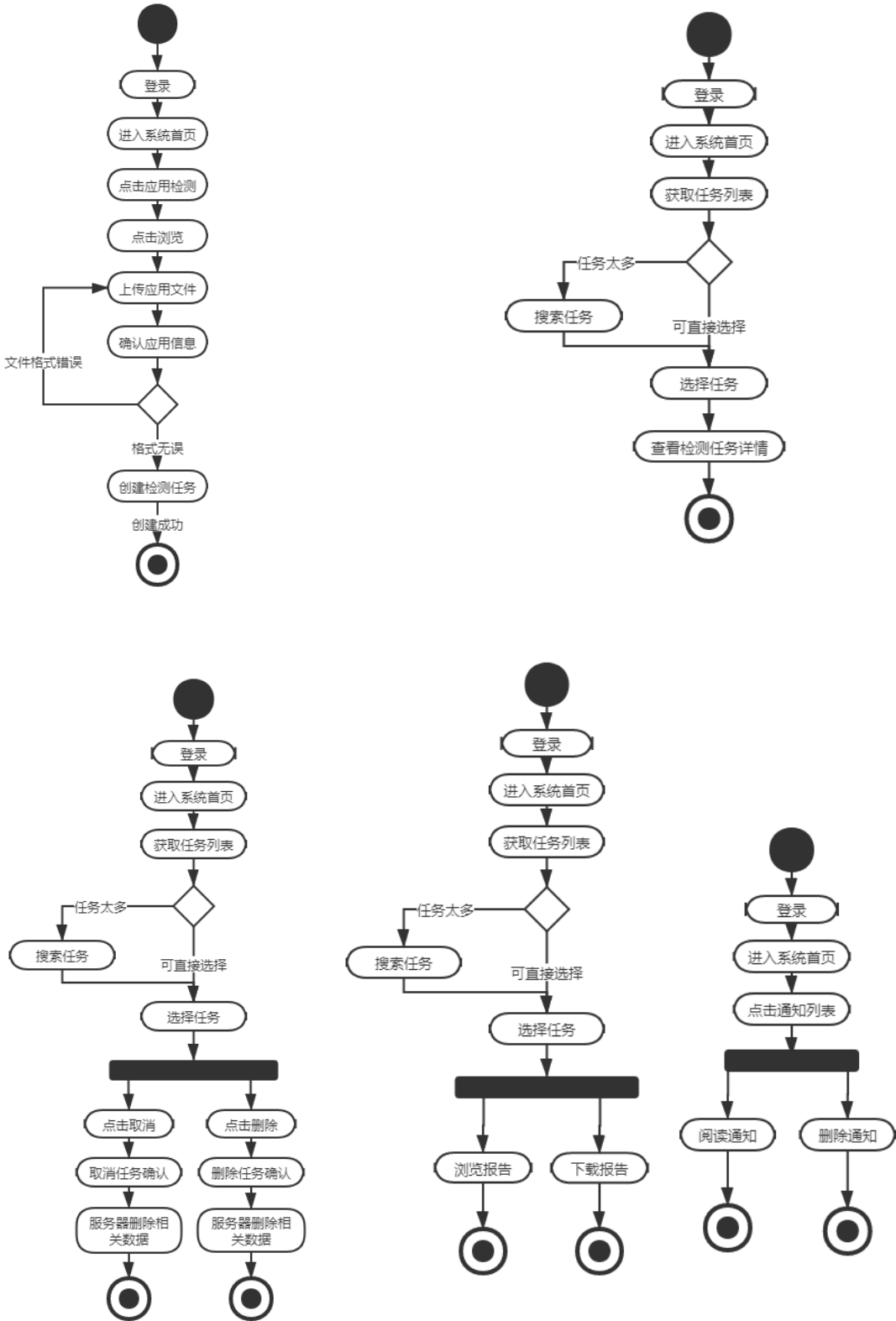
- (1) 其他闸机(例如高铁进站口闸机、图书馆进口闸机)或部分校门闸机只能单向通行,只允许进入,不允许出来。
- (2) 其他闸机口信息检测更加严格多样,例如高铁进站闸机口会进行人脸识别等。
- (3) 其他闸机口可能存在更加严格的警报装置,可能存在允许或禁止通行的屏幕文字图像提示。

第 2 题:

请绘制安卓灰色应用检测系统的任务管理模块活动图。
安卓灰色应用检测系统中检测任务管理模块的主要功能是创建检测任务、查看检测任务、取消检测任务、删除检测记录、报告管理和通知管理。以上任务都是在用户成功登录系统后进行。……



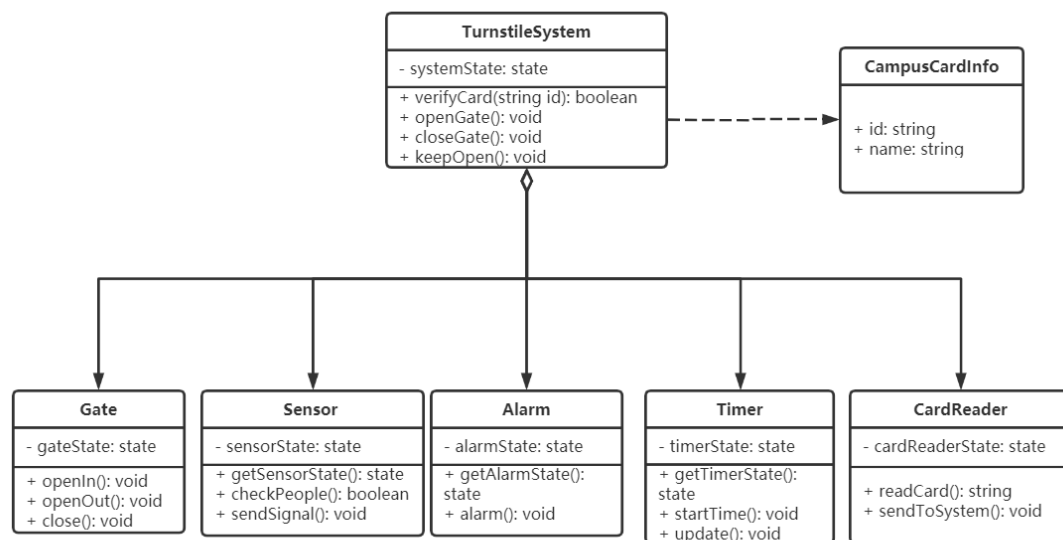
或者分开来画也可以



作业 3

第 1 题：

请仔细观察学校校门进出口闸机的工作方式，设计相关的类，画出相应的类图。



TurnstileSystem 闸机系统类：属性 `systemState` 表示状态，`verifyCard(string id)`根据读取到的校园卡 id 验证人员，`openGate()`打开闸门，`closeGate()`关闭闸门，`keepOpen()`当闸门开启时检测到有其他人将要通过或再次刷卡验证成功，刷新计时器保持闸门开启状态，可调用其他类的 `getSensorState()`方法获取传感器状态，`getAlarmState()`方法获取报警器状态，`getTimerState()`方法获取计时器状态。

CampusCardInfo 校园卡信息实体类：属性 `id` 和 `name` 存储校园卡 ID 和名字。

Gate 闸门类：属性 `gateState` 表示状态，`openIn()`向内打开闸门，`openOut()`向外打开闸门，`close()`关闭闸门。

Sensor 传感器类：属性表示状态，`checkPeople()`方法检测是否有人靠近，`sendSignal()`向闸机系统发送信息。

Alarm 报警器类：属性表示状态，`alarm()`方法当校园卡错误或有人尝试不刷卡尾随进入时发出警报。

Timer 计时器类：属性表示状态，`startTime()`当开启闸门后启动计时，`update()`当闸门开启时检测到内部有其他人将要通过或再次刷卡验证成功时刷新计时。

CardReader 读卡器类：readCard()读取校园卡信息，返回 string，sendTosystem()将读取到的信息发送给闸机系统进行验证。

TurnstileSystem 类与 CampusCardInfo 类是依赖关系，与 Gate 类、Sensor 类、Alarm 类、Timer 类、CardReader 类是聚合关系。

第 2 题：

请绘制安卓灰色应用检测系统的任务管理模块类图。

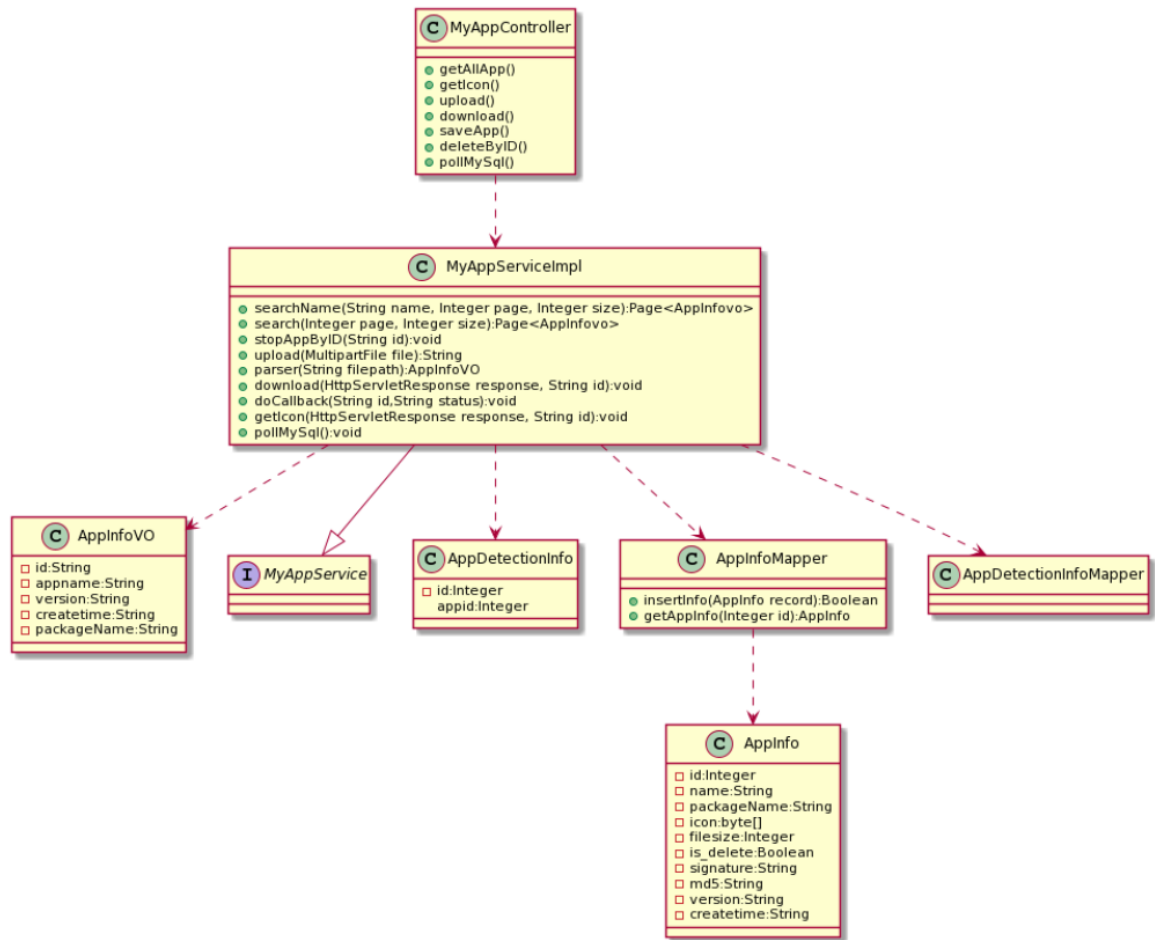
安卓灰色应用检测系统中任务管理模块的主要功能是创建检测任务、查看检测任务、取消检测任务、删除检测记录、报告管理和通知管理。**将该模块分为三部分：检测任务管理、报告管理、通知管理，请设计相关的类，完成该模块这三部分的类图。**

.....

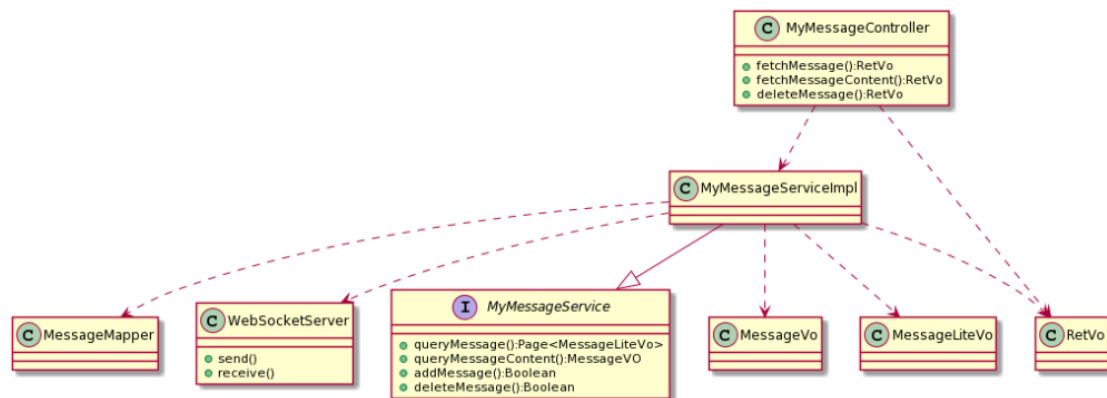
（附加：尝试画出检测平台的总体工作流程时序图，会有加分，但不作强制要求）

以下设计和类图作为参考即可

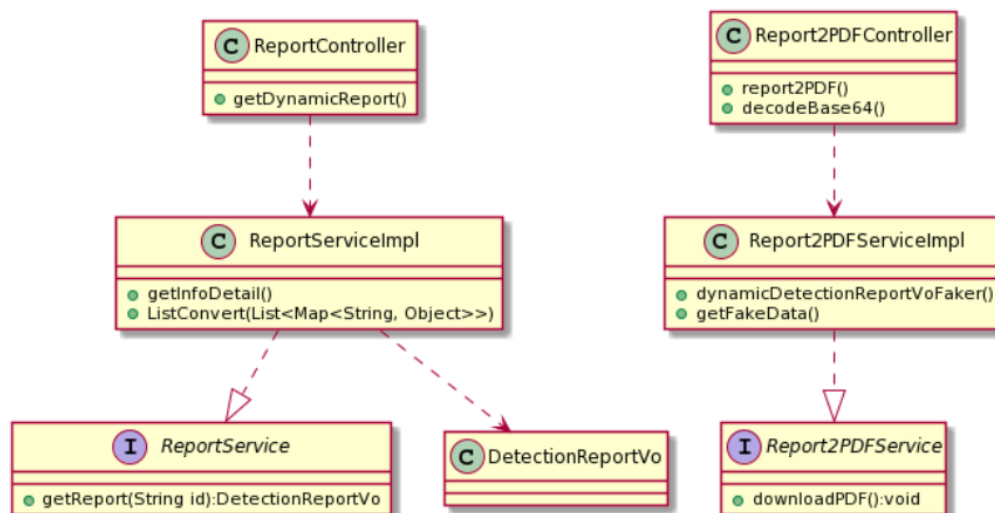
检测任务管理模块的主要类图如图所示，该模块的功能主要是创建、查看、取消检测任务，删除检测记录等。MyAppController 类负责用户上传应用请求并规定资源访问接口，MyAppServiceImpl 类负责请求逻辑中上传应用、查询应用、插入信息、保存文件功能的具体实现。AppDetectionInfoMapper 类和 AppInfoMapper 类功能相似，负责应用检测信息和应用信息的数据库查询和插入。AppInfo 和 AppInfoVO 是应用信息实体类和应用信息显示层对象。MyAppServiceImpl 类中的 upload()方法处理的是 Web 客户端安卓应用包文件的上传，APK 文件传输到服务端后保存在临时文件位置，返回临时 APK 文件的路径。parser()方法调用 upload()方法获得临时文件，调用工具获得应用的名称、版本和包名信息。用户确定创建任务后调用 saveApp()方法，该方法将服务端的临时文件拷贝到服务端存储应用文件的位置。getIcon()方法获取应用的图标。通过调用 AppInfoMapper 类中的 insertInfo()方法将返回一个 AppInfo 对象，将应用基本信息插入到数据库中。检测任务的调度通过 pollMysql()方法实现，该方法实现的是一个定时任务。



通知管理模块负责将检测任务结束通知和灰色应用检测通知推送给 Web 客户端，主要类如图所示。MyMessageController 类负责用户查看通知、删除通知的请求。MessageMapper 类用于从数据库查询删除消息数据。WebSocketServer 类中的 send() 方法用于向客户端发送消息，客户端和服务端的通信协议为 WebSocket 协议。MessageVo 类、MessageLiteVo 类、RetVo 类都是显示层对象，规定了 Web 端接收数据的格式。MyMessageService 是一个接口类，它的实现类是 MyMessageServiceImpl。MyMessageController 类收到新增消息时，对应的实现类 MyMessageServiceImpl 调用 addMessage() 方法，addMessage() 方法调用 MessageMapper 类将新消息内容插入到数据库中的 message_info 表。然后调用 WebSocketServer 类中的 send() 方法向客户端推送消息。WebSocketServer 类中的 receive() 方法用于接收客户端消息的请求，接到用户删除指定消息的请求，MyMessageServiceImpl 实现了 deleteMessage 方法，调用 MessageMapper 类从数据库中删除指定消息。



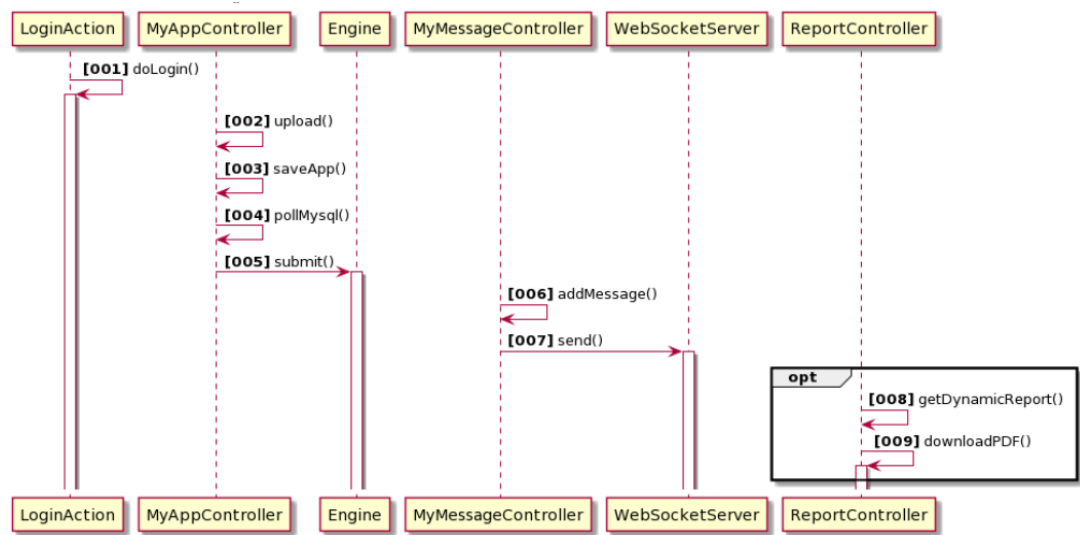
报告管理模块的主要实现了报告的浏览和下载功能，主要类图如图。ReportController 类和 Report2PDFController 类负责用户浏览检测报告、下载检测报告的请求。ReportService 类是接口类， ReportServiceImpl 是其实现类。DetectionReportVo 类是显示层对象，规定了 Web 端数据的格式。ReportService 类中的 getReport()方法从数据库和硬盘中获取数据，根据 Web 客户端规定的报告数据格式，将数据组装成 DetectionReportVo 实体类。同理，ReportDFServiceImpl 类是 ReportDFService 接口类的实现类。Report2PDFService 类中的 downloadPDF()方法根据模板生成 PDF 格式报告供用户下载。



总体流程

检测平台的总体工作流程时序图如图所示。用户登录平台时 LoginAction 类中的 doLogin()方法被调用，成功登录以后。用户上传应用 APK 文件，MyAppController 类

的 upload()方法负责上传应用，调用 MyAppServiceImpl 类的 upload()方法实现应用上传到服务端的功能。接着调用 MyAppServiceImpl 类的 saveApp()方法保存 APK 文件。调用 AppInfoMapper 类的 insert()方法将应用信息存储到数据库，接着调用 pollMysql()方法将检测任务加入到检测队列，pollMysql()方法调用检测引擎类 Engine 的 submit()方法向动态检测引擎提交 http 请求。检测任务结束时 MyMessageController 类调用 MyMessageServiceImpl 类的 addMessage()方法，然后调用 WebSocketServer 类中的 send()方法向客户端推送消息。检测引擎收到浏览报告的请求时，ReportController 类调用 getDynamicReport()方法，用户请求下载报告时，Report2PDFController 类调用 Report2PDFService 类的 downloadPDF()方法。



作业 4

第 1 题：

某计算机语言中规定，“标识符是由字母开头，后跟字母或数字的任意组合构成。有效字符数为 8 个，最大字符数为 80 个。不能是保留字”。请用等价类划分方法对标识符命名是否正确进行测试。要求给出等价类表，和具体的覆盖数据。

等价类表

需求	有效等价类	编号	无效等价类	编号
由字母开头	由字母开头	1	非字母开头	2
后跟字母或数字的任意覆盖数据组合构成	字母或数字的任意组合	3	含有非字母或数字的符号	4
有效字符数为 8 个, 最大字符数为 80 个	8~80 个	5	<8	6
			>80	7
不能是保留字	非保留字	8	保留字	9

覆盖数据

测试用例编号	输入	预期输出	用例描述	覆盖等价类
1	A123456789	OK	正常	1,3,5,8
2	123456789	Error	非字母开头	2,3,5,8
3	A12#456789	Error	还有非字母或数字的字符	1,4,5,8
4	A123456	Error	少于 7 个	1,3,6,8
5	A12345678901234567890 12345678901234567890 12345678901234567890 12345678901234567890	Error	大于 80 个字符	1,3,7,8
6	unsigned	Error	保留字	1,3,5,9

第 2 题:

给出以下两个代码的**环路复杂度**, 并给出所有的**独立路径**, 同时对于每条独立路径给出**完整测试用例**以及对应输出。

代码一:

<pre> 1 int i = 0; 2 int n = 4; 3 while (i < n-1) { 4 j = i + 1; 5 while(j < n) { 6 if (A[i] < A[j]) { 7 swap(A[i],A[j]); 8 } 9 j = j + 1 10 } 11 i = i + 1; 12 }</pre>
--

代码二:

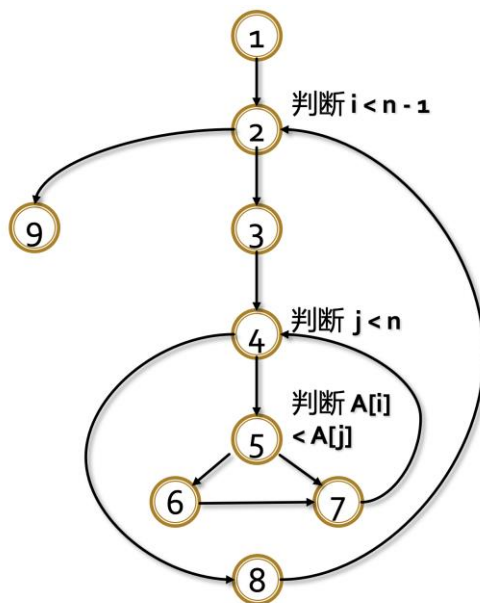
```

1 public int sum(int n, int upperbound) {
2     int result, i;
3     result = 0;
4     i = 0;
5     if (n < 0) {
6         n = -n;
7     }
8     while(i<n && result <= upperbound) {
9         i = i + 1;
10        result = result + i;
11    }
12    if(result <= upperbound) {
13        System.out.println("The sum is " + result);
14    }
15    else {
16        System.out.println("The sum is too large!");
17    }
18    return result;

```

(1)

流图如下所示（流图中的结点标识最好能说明代表程序中的哪些代码段，或者用相应代码行号作为结点标识，这里只简略说明了判定结点）：



环路复杂度 $V(G) = 11$ (边数) $- 9$ (结点数) $+ 2 = 4$

Path[1]: 1,2,9

Path[2]: 1,2,3,4,8,2,9

Path[3]: 1,2,3,4,5,7,4,8,2,9

Path[4]: 1,2,3,4,5,6,7,4,8,2,9

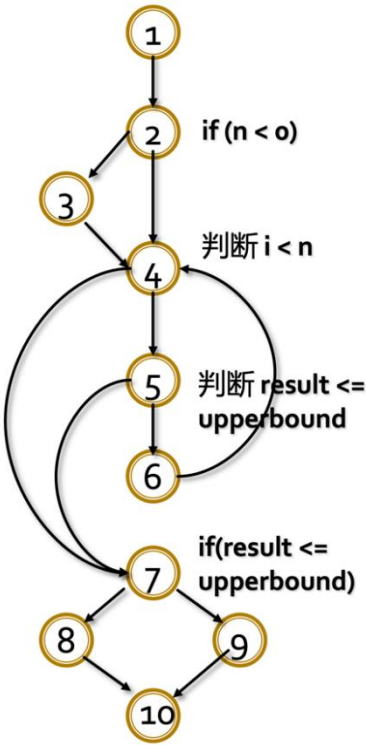
若代码中存在循环，可将循环体视为只执行一次，这样方便写独立路径。

测试用例不唯一，满足路径即可，对于实际不存在的路径，进行说明后可不进行测试。

例如对于路径 2: 1-2-3-4-8-2-9，结点 4 判断 $j < n$ ，而 $j = i + 1$ ，当 $i < n - 1$ 成立时， $j < n$ 必定成立，结点 4 绝不会跳到结点 8，故该路径实际不存在。

独立路径	测试用例	结果
路径 1: 1,2,9	$i = 4, n = 4, A = [\dots]$	数组 A 元素顺序不变
路径 2: 1,2,3,4,8,2,9	\	\
路径 3: 1,2,3,4,5,7,4,8,2,9	$i = 0, n = 4, A = [4, 3, 2, 1\dots]$	数组 A 变为 $[4, 3, 2, 1\dots]$
路径 4: 1,2,3,4,5,6,7,4,8,2,9	$i = 0, n = 4, A = [1, 2, 3, 4\dots]$	数组 A 变为 $[4, 3, 2, 1\dots]$

(2) 流图如下所示



环路复杂度： $V(G) = 4$ (判定结点数) + 1 = 5

Path[1]: 1,2,4,7,8,10

Path[2]: 1,2,3,4,7,8,10

Path[3]: 1,2,4,7,9,10

Path[4]: 1,2,4,5,7,8,10

Path[5]: 1,2,4,5,6,4,7,8,10

测试用例不唯一，满足路径即可。测试用例及结果略……