Terminator整理

实习内容：负责开发ios客户端Terminator,基本功能是管理ios设备上的国际化内测包的安装、更新和应用跳转功能。获取网络数据，在本地获得应安装应用的bundleID、版本信息、更新时间，然后结合网络和本地数据获得显示数据，最后在界面进行显示。

实现难点:在IOS10以及之前版本，可以通过私有api MobileCoreServices获得所有的安装应用、版本信息、更新时间，实现较为容易。在ios11以及之后的系统，由于私有api被封，一度陷入困境。后面受一个砸壳appstore实现appstore自动下载的案例的启发，通过砸壳可以appstore经过代码混淆过的部分，通过研究appstore如何获得本地安装应用的信息，发现是在本应用中添加了权限之后才可以使用。

关于如何使用私有api进行在appstore上线，这里由于使用了类似java反射的技术来使用私有api，目前appstore检测私有api的使用是通过静态检测私有api是否使用，因此将名称以uft-8的形式进行保存，成功绕开了上线检测。当然如果苹果官方以后采用动态检测的方式来检测私有api，则需要考虑其他的方法绕开检测。

功能优化：在刷新时做了加载优化，首先显示界面的外形，然后在数据处理之后再进行了图片加载；在左右滑动优化，当滑动页面多于3个时，可以进行优化，每次只加载3个页面的数据，如果页面已经加载则不进行加载，当滑动结束之后判断左右界面是否加载，如果没有加载则进行加载。