User Guide for *HackSpace Mobile*

1 Application installation and environment configuration

1.1 Apk installation and testing

For quick usage, I recommend you to install the HackSpace_Mobile.apk in your Android devices (version Android 5+). The apk file was compiled by Ionic tools. After successful installation, the application can be used in your Android device.

1.2 Testing by Ionic server

If you want to test the application by Ionic server on the laptop or desktop computer, please follow these steps:

1.2.1 Install node.js

You can download the installation package from https://nodejs.org/en/download/

1.2.2 Install Cordorva and Ionic

If your operating system is Window or Linux, please execute the command in command line tool:
$ npm install –g cordova ionic
If your operating system is Window or Linux,, please execute the command in command line tool:
sodu npm install –g cordova ionic

1.3 Test the application

Now enter the root directory of the application:
If you want to test it by Chrome, input the command. Then the Chrome view page will be activated:
$ inoic serve
If you want to test it by Android device, input the command:
$ ionic platform add android
$ ionic run android
If you want to test it by IOS device, input the command:
ionic platform add ios
ionic run ios

2 Demonstration

Important: To test the system, I recommend you to use **two or more** devices (Android or Windows) for better experience because the functions are mainly about location-based service. Feel free to sign up or login through any defined username. However, if you want to register the test devices in *HackSpace Mobile*, please use **the same** username to login

and register the devices to ensure that they are bound to the same user. Otherwise you may need another device to test the system because a user can only trace the devices that belong to (bind to) him or her.

Notes: Some of the screen shots are based on Window 8 and Chrome. As a result, the effects could be a little different from Android platform.

2.1 User Signup and Login

The users can sign up by a pair of username and password. The register will fail if someone else has registered the same username. The users can successfully login through correct username and password.

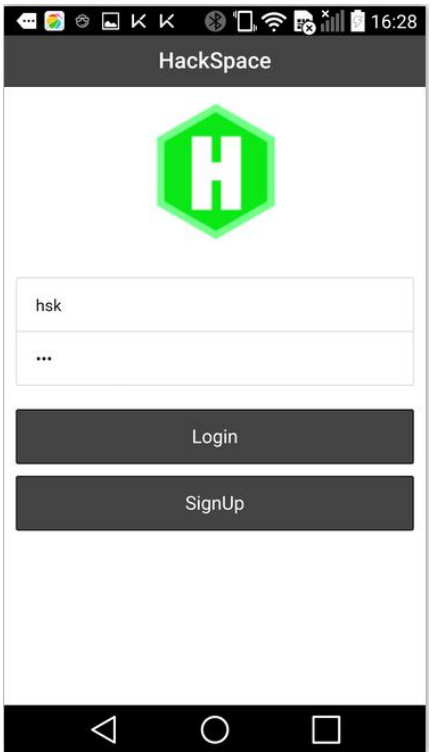


Figure 1 Login

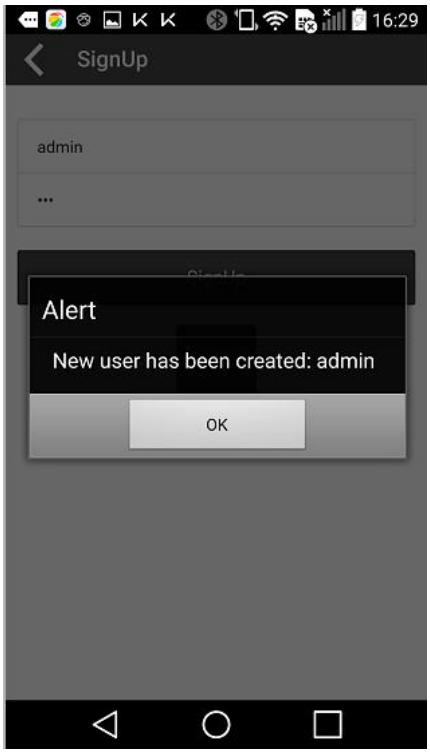


Figure 2 Sign up

2.2 Register your device

Let's see from Figure 3, there is nothing on the "Your Devices" interface. By clicking the button on the top right corner, we can register the device and bind it with a unique code and the username. It should be noted that the first user who register the device will be the 'owner' of the device. The system does not allow the same device to be registered more than once.

Now you are the owner of the device you are operating on! If you want to **change the owner of the device**, the steps are: 1. Click your target device and choose "Logoff the device" (as shown in Figure 6); 2. Click 'Setting' tab on the bottom and logout you account; 3. Login with a new username; 4. Register again the device. By following the steps the device will be registered and bound with a new user.

Notes: If the device is registered as ‘Private’ device, it can only be viewed or traced by its owner. If the device is set as ‘Public’, it can be operated by all the users.

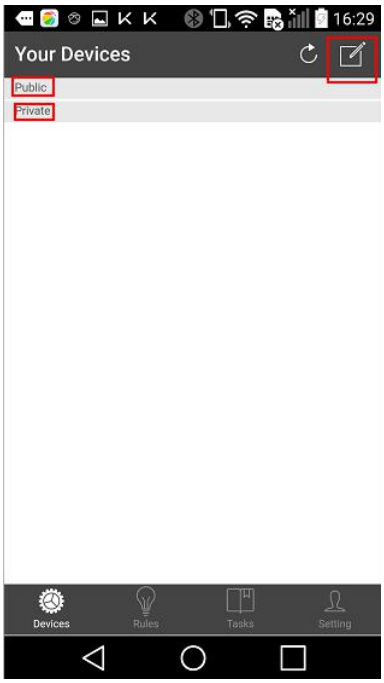


Figure 3 “Your Devices” Interface

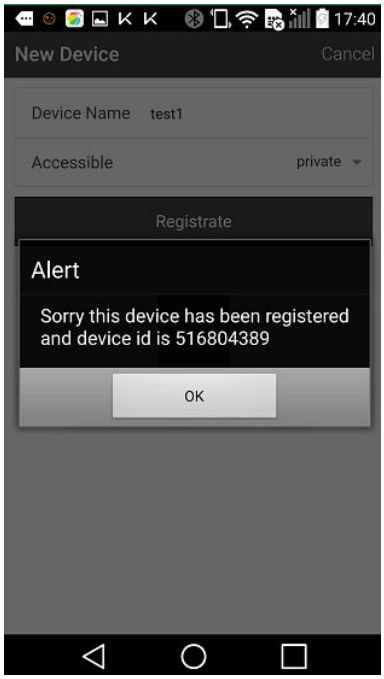


Figure 4 Fail to register the second time

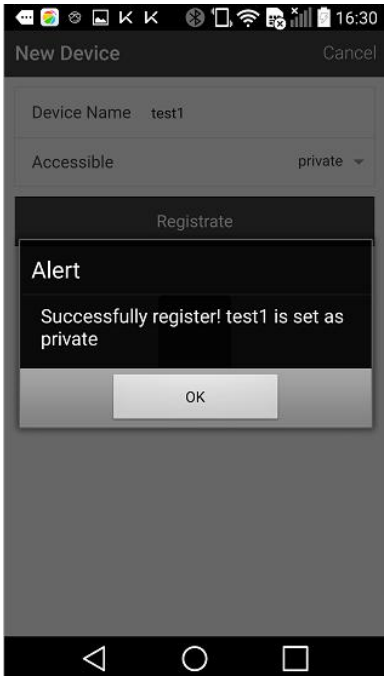


Figure 5 Registering the device

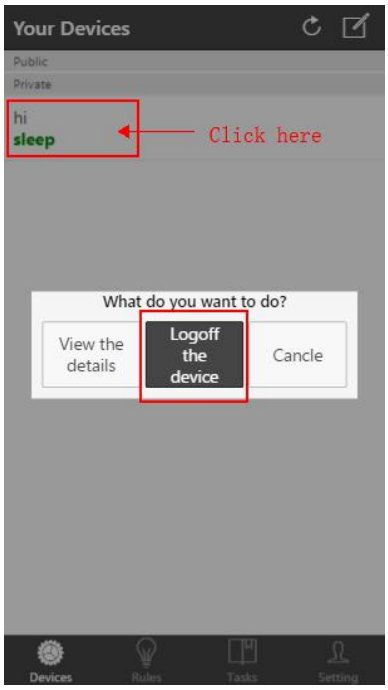


Figure 6 Logoff the device

2.3 Viewing your devices

A user is allowed to register multiple devices. In the ‘Devices’ interface, you can view all your private devices and all the public devices. As shown in Figure 6, you can see the device state under each device. The state will be red-letter ‘activated’ when the device is

uploading its geolocation data. Oppositely, the green-letter state ‘sleep’ means that the device is not uploading its coordinates.

Notes: Only the ‘activated’ device can be traced in user rules. To keep a device that is ‘activated’, you need to go to ‘Setting’ interface and turn on the toggle. The details can be seen from Figure 7.

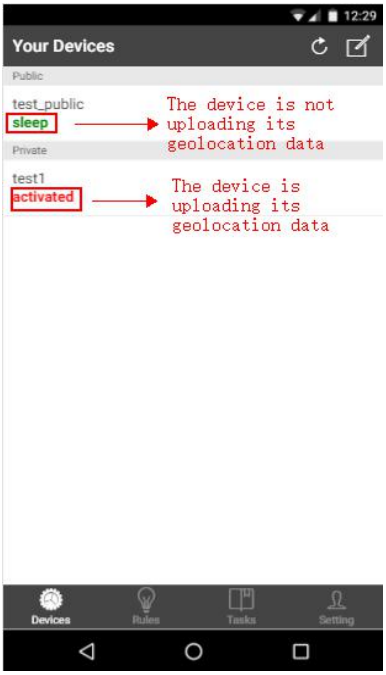


Figure 7 ‘Device’ interface

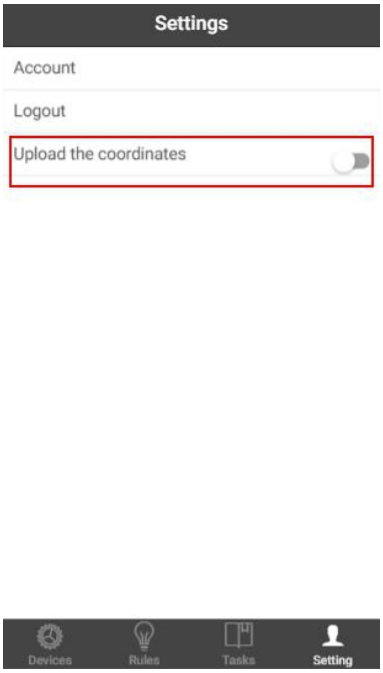


Figure 8 Activating a device

2.4 Customize the rules

The core function of *HackSpace* is rule customization. In the “Rule” interface, users can choose different components from select boxes to compose a rule. For example, in the first select box, if the user chooses ‘absolute_geolocation’ as the data type, he/she is able to pick an absolute location on the map and click “save” to decide the location. Then the user can choose operations, numbers, output and action to design the rule. As shown in Figure 7, the rule can be described as “IF the device is more than 1000 meters away from an absolute location, THEN the screen alerts an alarm”. When the rule is set up well, you can click the right-top button ‘Hack It!’ to register your task.

Notes: Before designing your rules, please allow the app to use location service. The steps are: Setting -> General -> Location -> Location Service On.

The select box ‘DataType’ means the type of location-based service. If your choose ‘absolute_geolocation’, the system will provide geo-based memo or geo-fencing service. If you choose ‘moving_geolocation’, the system will offer geo-tracing service to track a device. The select box “Logic Gate”, ”Date”, “Time” can not be chosen yet because they are irrelevant to location-based service.

Figure 9 Customizing a rule

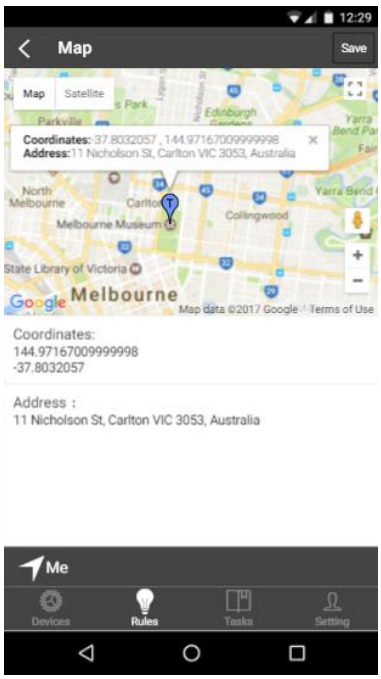
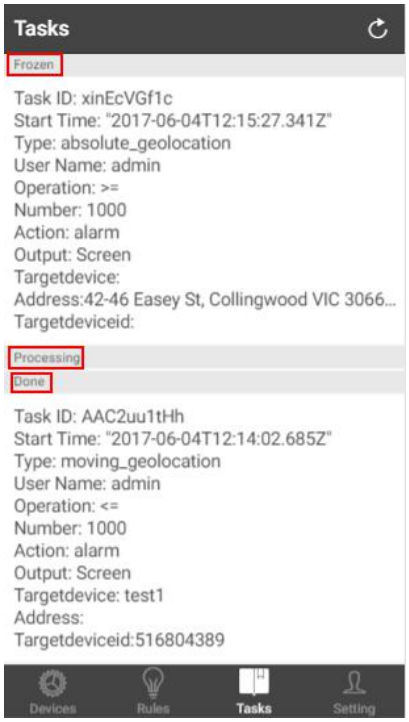


Figure 10 Picking an absolute geo location

2.5 View your tasks and process them

Once the rule is designed, it will be encapsulated as a ‘frozen’ task. Let’s view the tasks in ‘Tasks’ interface. The tasks that are just defined by rules will be set as ‘frozen’. Click each task and choose to process the task or delete it. If you choose ‘Process the task’, the task will be push to ‘Processing’ queue. The tasks will be processed automatically in ‘Processing’ queue. You are also able to operation the tasks in ‘Processing’ queue by stop them or delete them. If you choose to stop a ‘processing-task’, it will be pushed to ‘frozen’ queue and wait for process.



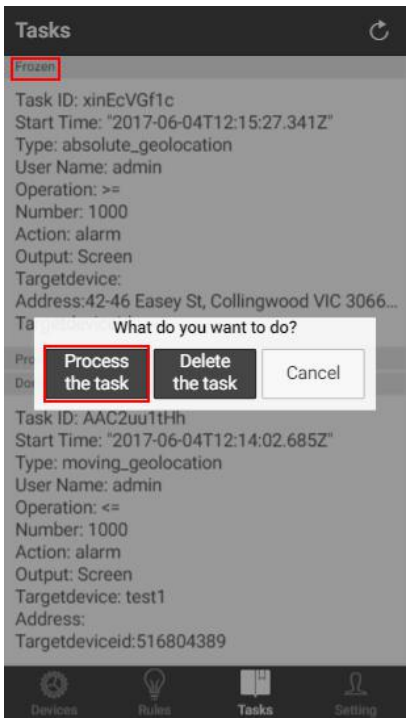


Figure 11 Task interface

Figure 12 Processing a frozen task

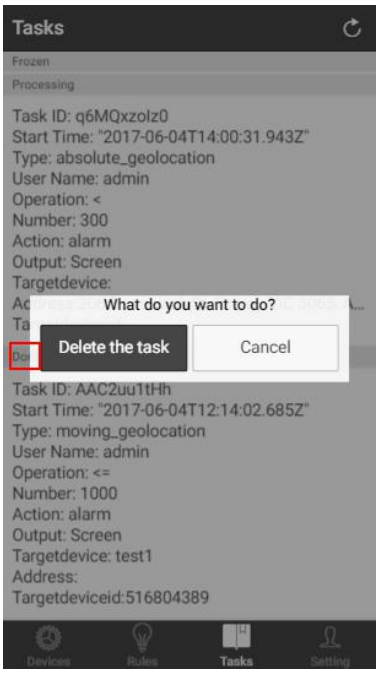


Figure 13 Stop or delete the tasks being processed

Figure 14 Delete the task that has been done

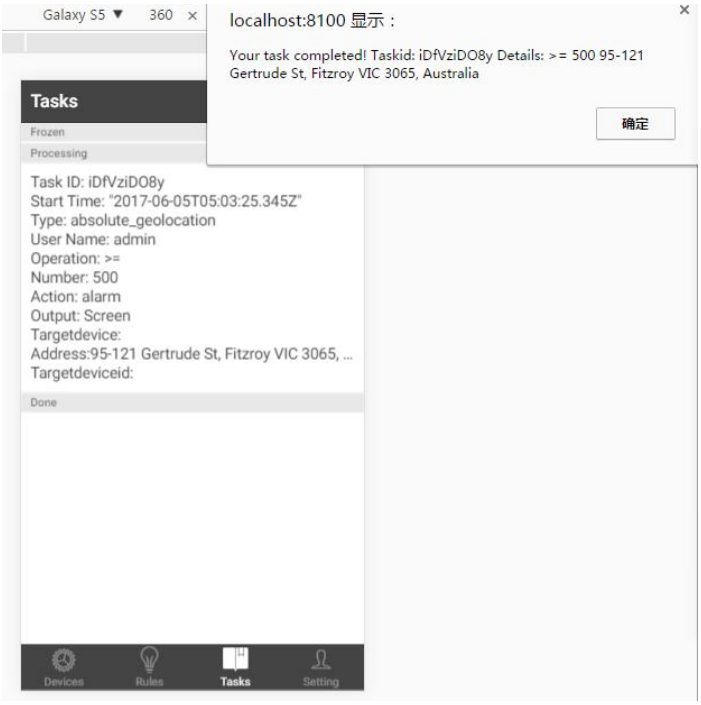


Figure 15 The task is completed and the screen gives out an alarm.

3 The problems and solutions

There is a risk that the background process would be killed by Android. There are various approaches to avoid the application to be killed by process manager. For example, I can disguise the application and make it like a system-wide application to get away with process-killing. It is

also possible to create double-process services to protect each other, which means that if one of the services is cleared, the other will immediately reset the previous one. However, these methods above act against a good ecological environment for Android.
A better solution in the system is that in ‘Tasks’ view, click a processing task and choose ‘Stop the task’ to freeze the task. Then you can activate the task again by click the same task which has been frozen and choose ‘Process the task’. Following the steps, your task will be executed well.

4 Contact

If some parts of the guide are difficult to understand, please contact the developer. I will also appreciate it if you find any bugs and submit the bugs to me.
Email: shikaih@student.unimelb.edu.au