

Alarms and Networking

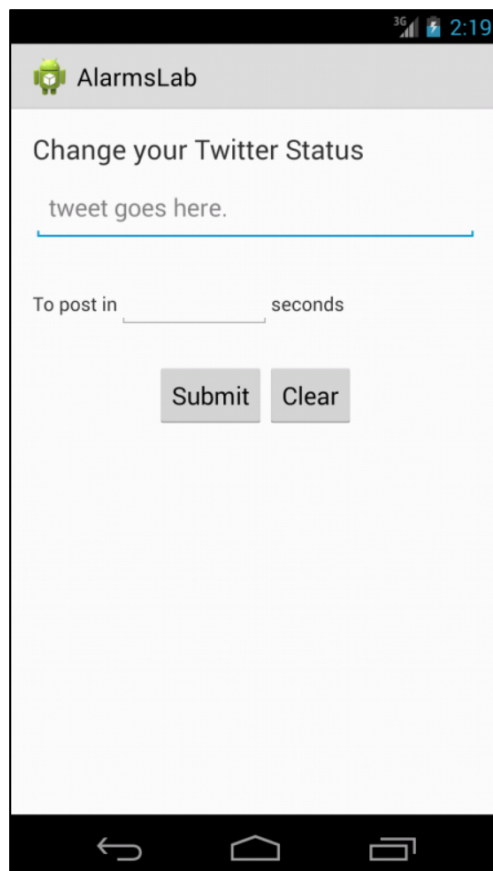
Objectives:

Familiarize yourself with Alarms and Networking. Create an application that uses Alarms, Networking, and AsyncTasks.

Once you've completed this lab you should have a better understanding of Alarms, Networking and AsyncTasks. You should know how to use and create alarms using the AlarmManager Service, how to use Networking support classes to send HTTP POST/GET requests to other services on the Internet, and how to use AsyncTasks to perform short tasks on a separate thread.

Overview:

Using the app depicted below, the user will enter a Twitter status update and then set a delivery time, measured as some number of seconds from the current time. After this, if the user hits the submit button, the code should set an Alarm to go off at the specified time. When the Alarm goes off, it should



start the AlarmTweetService, which will do the work of posting the new status to Twitter. You can see your posts on the www.twitter.com website.

See the screencast of the app in operation.

Implementation Notes:

- 1) Grab the started code you need as you have been doing all semester long your upstream repo. As a reminder, the commands to execute from inside your local repo are the following:

`git fetch upstream master`

`git pull upstream master`

`git push origin master`

At this point you should see a new directory inside of your repo named “Lab7_Alarms”. Go ahead and open up this directory with Android Studio as an existing Android Studio Project.

- 2) Modify the code near the top of the AlarmTweetService so that not everyone is using the same Twitter Account. Note that since, we will be sharing several Twitter accounts, you should put some unique identifier on your tweets. Otherwise, you won’t be able to distinguish your tweets from those of your classmates. Also modify AlarmTweetService to use an HttpURLConnection.
- 3) Modify AlarmCreateActivity, paying close attention to the TODO items. Some of the TODO’s give you directions for the code to implement but others ask questions that may appear on an exam.

Submission:

Submit in our normal way by committing your changes and pushing to your repo. If possible try and remember to use the commit message “completed Lab7_Alarms” in your final commit so we know which commit best represents your final submission.