**README**

A chord based distributed hash table (DHT) built using Android as the distributed systems. All node joins are handled in realtime and position in the rings are alloted by the first avd, avd0. Built as a part of the CSE 586 Distributed Computers course at University at Buffalo.

**Usage**

* Create the 5 AVDs using the create\_avd.py script: python2.7 create\_avd.py 5 /PATH\_TO\_SDK/
* Run the 5 AVDs using the run\_avd.py script: python2.7 run\_avd.py 5
* Run the port redirection script using: python2.7 set\_redir.py 10000
* Now, run the grading script as follows: ./simpledht-grading.linux app-debug.apk
* The simpledht-grading.linux script is present in app/build/outputs so give the path accordingly

For more information about the grading script, run ./simpledht-grading.linux -h

**Implementation**

* Implemented three things: 1) ID space partitioning/re-partitioning, 2) Ring-based routing, and 3) Node joins.
* The content provider should implement all DHT functionalities and support insert and query operations. Thus, if you run multiple instances of your app, all content provider instances should form a Chord ring and serve insert/query requests in a distributed fashion according to the Chord protocol.
* The implementation details of each method i.e. insert(), delete(), query() and serverQuery() can be found in the code.