ASH CLOUD

Andy Alexander, Mark Capobianco, Jinfeng Lin, Ruoyun Song

http://nanotwitter.herokuapp.com/
https://github.com/Ash-cloud/nanotwitter

Progress Made 3/30/15

- Fixed bugs
    - Hyperlink on unlogged in root page lead to unlogged user post twitter and cause crash on loggedin page.
    - Redirect to logged in root from Unlogged in root if session status = true
    - Display user page button based on session status (Register/Login vs My Page/Follow)
- Start design UI
- New features
    - Show user name on the logged in root page, for user to follow.
- Preparing for REST API
- Imported Seed Data
    - Import part of the seed data, due to the limitation of heroku database size.
- Instrumented New Relic
- Completed I/O Loader Tests for GET of unlogged root page.

TESTS

GET of Unlogged In Root - Maintain Client Load
100 Clients over 1 min
http://ldr.io/1a8dZh5

Stable for 1 min

GET of Unlogged in Root -Maintain Client Load -
250 Clients over 1min
http://ldr.io/1MpMcJQ

Currently support 250 clients for 1 min. Introducing timeouts close to 1 min

GET of Unlogged In Root - Client per Second
100 Clients over 1 min
http://ldr.io/1HfNaSb
Could not handle 100 clients per second very well. Aborted due to error threshhold being reached at around 12 seconds

GET of Unlogged in Root - Client per Test
100 Clients over 1 min
http://ldr.io/1CqhBUw
Successful completion of this test. No timeouts. Average response time 146ms.

GET of Unlogged in Root - Client per Test
250 Clients over 1 min
http://ldr.io/1Esq6SN
Success, no timeouts. Average response time 173ms.

GET of Unlogged in Root - Client per Test
500 Clients over 1 min
http://ldr.io/1MpQH75
Success, no timeouts. Average response time 306ms.

Get of Unlogged in Root - Client per Test
1000 Clients over 1 min
http://ldr.io/1CqjkJB
Success, no timeouts. Average response time 408ms.

Summary:

In our test we conducted a test suite with 7 tests inside, we firstly applied 3 categories of tests to our welcome page, and do more with larger quantities of users to see the performance. We firstly verified the wrong site which is nanotwitter.heroku.com, this site will redirect the request to nanotwitter.heroapp.com thus the requests are abort at middle make the illusion that our server is strong enough to hold a large amout of user. Later we modified it the correct address, and show the results above.

In the graph we could find the limitation of user amount is about 250 or less on the persistence test. We are not scaling well right now, and the welcome page will keep retrieving database to create timeline for every refresh. Definitely, we could apply caching at this place to improve the performance.