

Count Me Up

March 2017

Created by Carmelo Duchetta 11/03/2017

ARCHITECTURE

The architecture of the project supports a distributed environment using the Web Services development to take advantage of all the capabilities and resources of distributed computing in the most efficient manner possible.

The architecture could be deployed in different servers to make the application completely scalable.

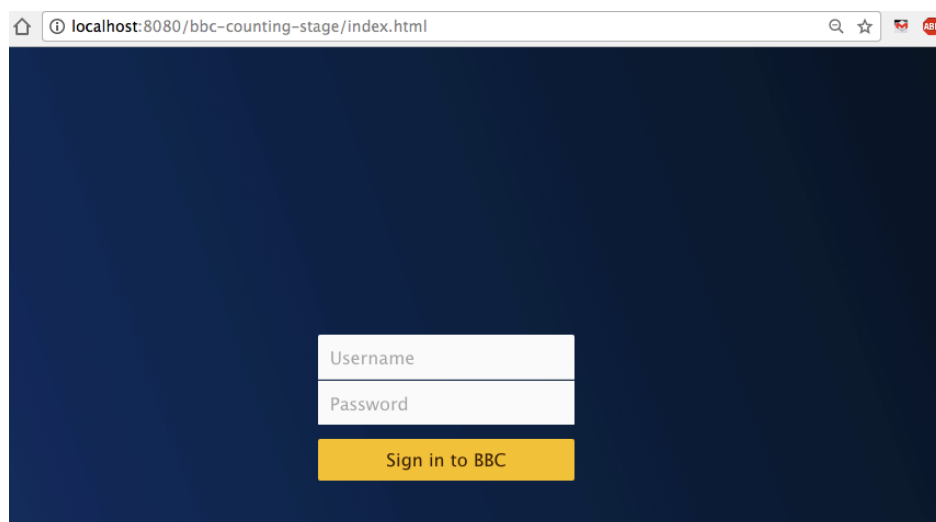
One PHP server that provide the interface with the external world and all users can interact with the application's functionalities.

J2EE is the main core technology used to design the Entities of the system. At the last layer there is Cassandra NoSQL Database. The choice of this database is related to different reasons as Write Speed, Multi-DC Replication, Tunable Consistency, JVM Based, CQL language. Also Cassandra DB is one of the best candidate to be integrated with Apache Solr technology used as well in this project to guarantee high performance in showing results. That project has been tested just on a single machine locally.

FUNCTIONALITIES

1. LOGIN
2. USER REGISTRATION
3. DASHBOARD
4. VOTE
5. SHOW RESULTS
6. LOGOUT

1. Login

A screenshot of a web browser window showing a login page. The browser's address bar displays 'localhost:8080/bbc-counting-stage/index.html'. The page has a dark blue background. In the center, there is a white login form with two input fields: 'Username' and 'Password'. Below these fields is a yellow button with the text 'Sign in to BBC'.

After the user has created his own profile, he will be able to login on the system.

2. USER REGISTRATION

localhost:8080/bbc-counting-stage/add-user.php

User Information

User details and requirements
Please enter your user details below and save the form

Title

Firstname

Surname

Full Name

Company / Individual

Email

Landline

Mobile

Login Info

Username

Password
Password (Minimum 6 characters)

Confirm Password

Save

Cancel

Each user will be able to vote after successfully registration compiling that form provided by a public link. (Local Public Link: <http://localhost:8080/bbc-counting-stage/add-user.php>)

3. Dashboard

localhost:8080/bbc-counting-stage/welcome.php

Logout

Welcome back! What would you like to do?

Show Results

Vote

The user will be able to choice in Voting or Showing in real time the results.

4. VOTE

localhost:8080/bbc-counting-stage/add-vote.php

Logout

New Vote

Note (*):

Please not exceeded the maximum allowed number of votes. Maximum of 3 times.

Votes

Please vote individually. After 3 times the vote will be ignored.

Candidates

CANDIDATE 1

Vote

Cancel

Each user can vote one of 5 Candidates. They are just static info to easily manage the data into Cassandra Database and Solr Core. After 3 times the user's vote will be ignored but it will be recorded on the database related to a Default User.

5. RESULTS

localhost:8080/bbc-counting-stage/show-results.php

Logout

Results

Date: 2017-3-11 2017-3-12 Real Time

	Candidate 1	Candidate 2	Candidate 3	Candidate 4	Candidate 5
This Period	3	0	0	0	0
Total	9	0	0	0	0

The user will be able to select a range dates and clicking through on Real Time button will be able to see the results for each candidate. Also the user can log out from the system clicking through the Logout button at the top corner on the right.

Missing Implementations with Solutions

- To show every second the results we could implement a javascript loop using ajax invoking the same function as the Real Time button. In that case we'll update that screen every second.
- The date range selection, provided in the showing results page, supports for now just date per day. Assuming the competition is just one day, that selecting date should supports time per hours and seconds.
- To test the showing results function within 1 sec, in term of performance, I thought to generate a script in Java to populate the Solr Schema with millions (10M) of records and to use JMeter to graph the response time. I'm assuming that Apache Solr Technology will provide high performance to satisfy that requirement.