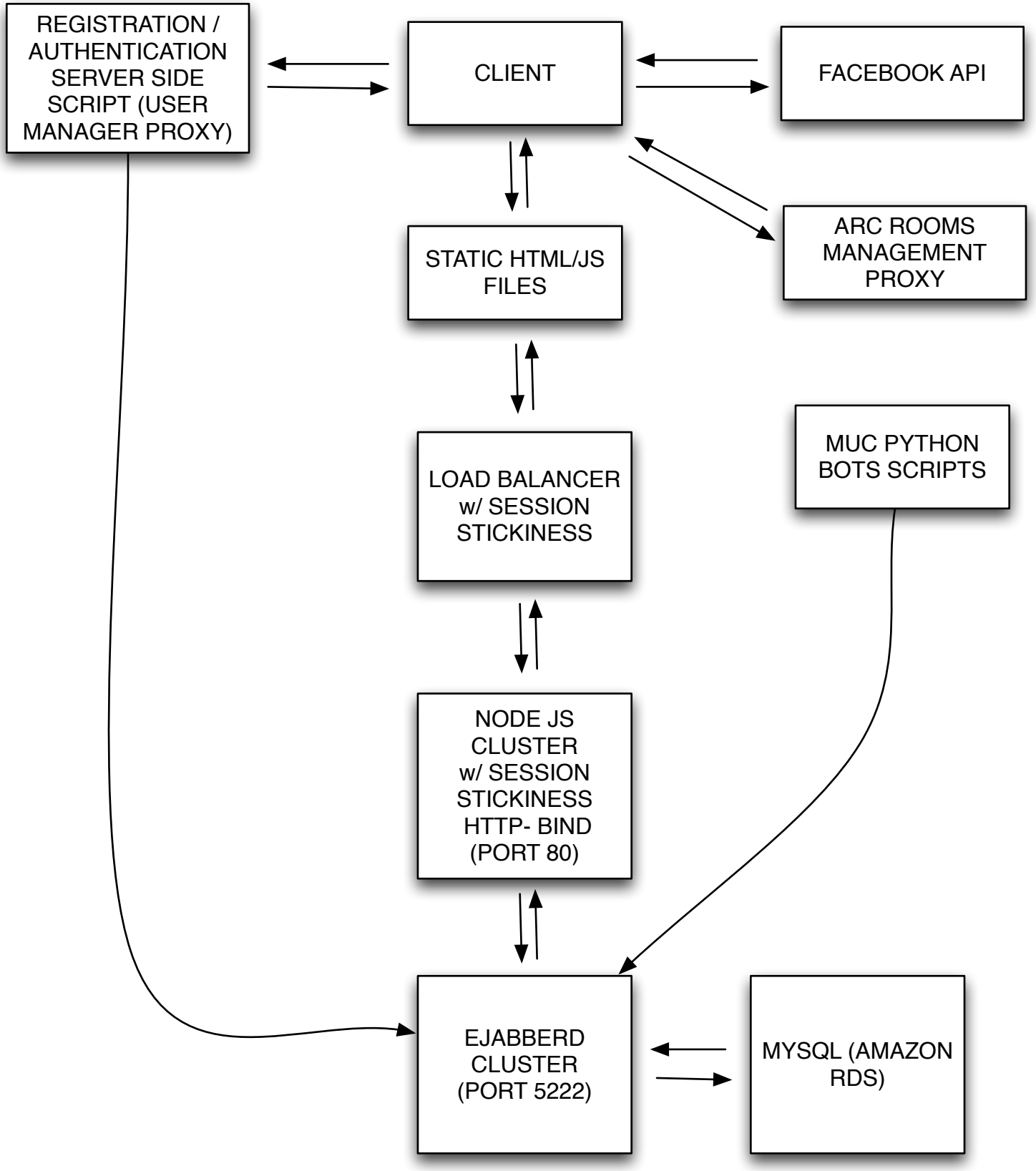
CHAT SERVICE



PUBLIC CHAT

2 USER ROLES

VISITOR

- NOT AUTHENTICATED ON FACEBOOK (ANONYMOUS)

- CAN VIEW CHAT ROOM MESSAGES

- CANNOT POST MESSAGES

- CANNOT HAVE PRIVATE CONVERSATIONS WITH OTHER MEMBERS

PARTICIPANT

- AUTHENTICATED ON FACEBOOK

- CAN VIEW CHAT ROOM MESSAGES

- CAN POST MESSAGES

- CAN HAVE PRIVATE CONVERSATIONS WITH OTHER MEMBERS

MUC BOT

- THE MUC BOT WILL BE AN AUTOMATED AGENT, RESPONSIBLE TO SET THE USER'S ROLE WHEN THE USER ENTERS A CHAT ROOM

- IF THE USER IS AUTHENTICATED ON FACEBOOK AND HAS ITS VCARD NICKNAME PROPERTY SET AS ITS FACEBOOK ID, THE BOT WILL GRANT THE USER THE STATUS 'PARTICIPANT' .

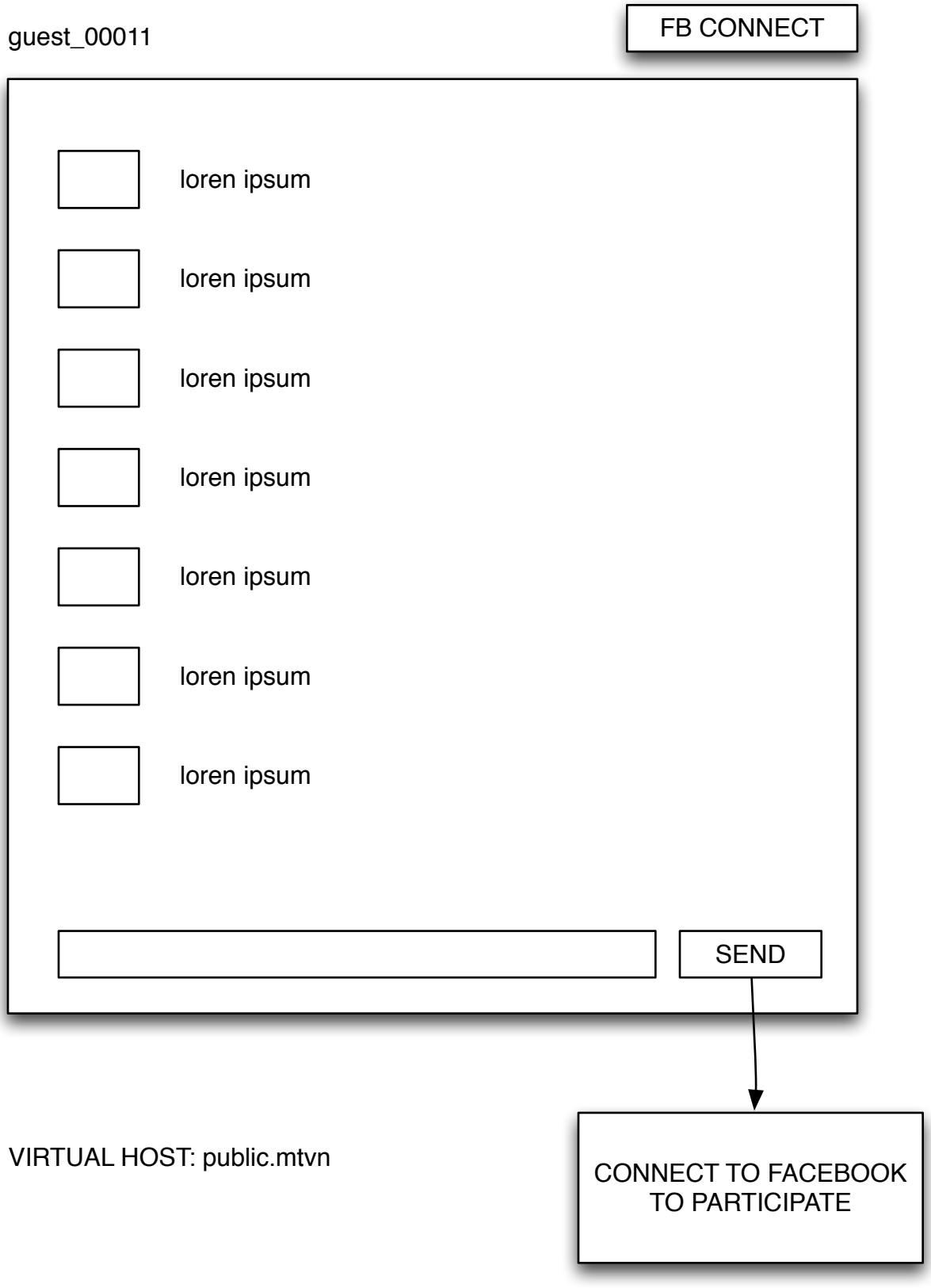
- IF THE USER IS NOT AUTHENTICATED, THE BOT WILL GRANT THE USER THE STATUS 'VISITOR'.

- EACH ROOM SHOULD HAVE A MUC BOT

- IT'S POSSIBLE TO HAVE ONE MUC BOT FOR MULTIPLE ROOMS, BUT THIS WOULD GIVE THE BOT A POTENTIALLY IMMENSE ROSTER LIST, WHICH COULD NEGATIVELY AFFECT THE SERVER PERFORMANCE.

- ONE ALTERNATIVE IS TO DEVELOP A CUSTOM ERLANG COMPONENT THAT WILL BE PLUGGED IN INTO THE EJABBERD SERVER.

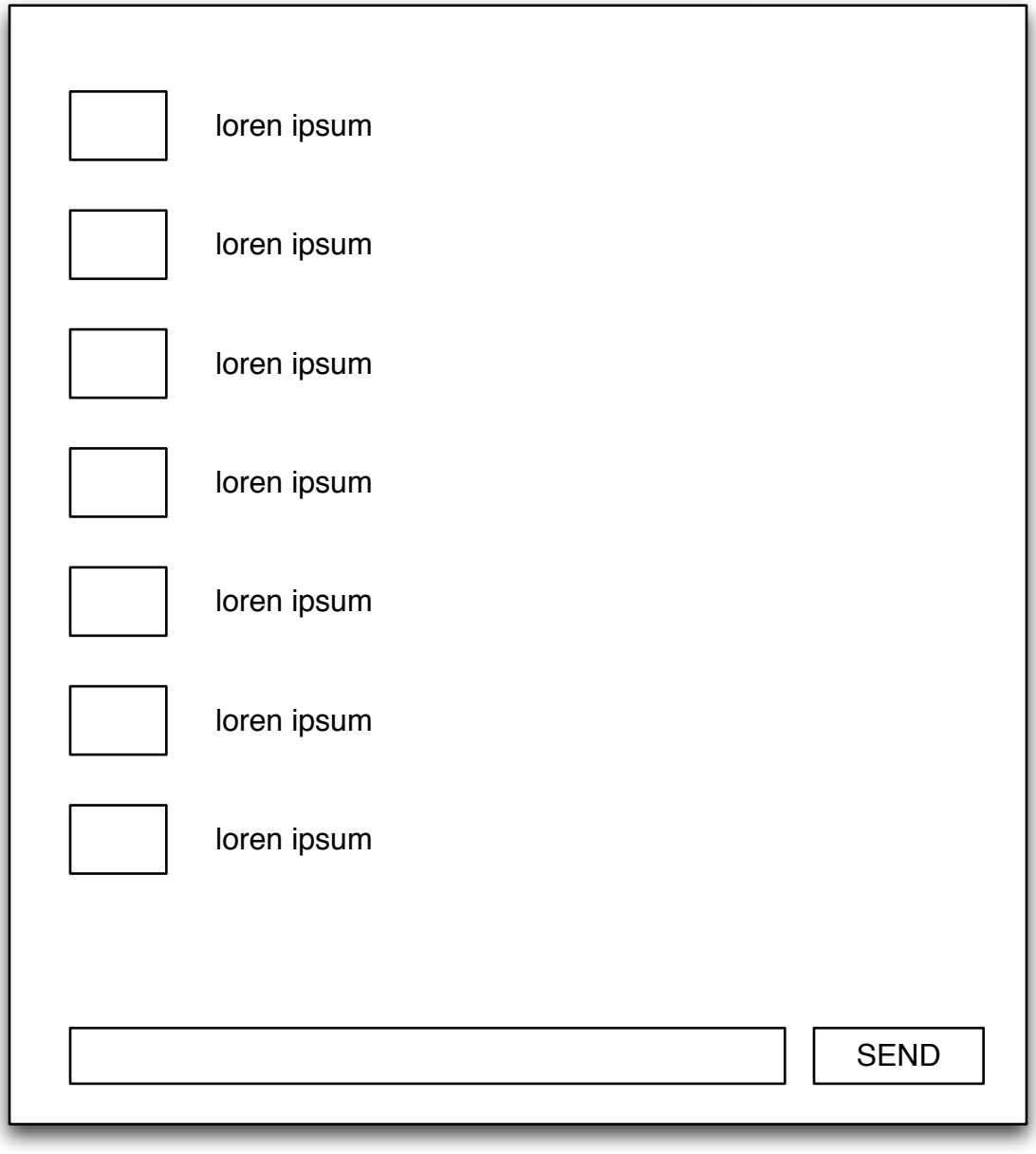
ROLE: VISITOR (Cannot send messages)



ROLE: PARTICIPANT (Can send messages)

ISAAC SILVA

CONNECTED



VIRTUAL HOST: conference.MTVN

USER MANAGER PROXY

CURRENT FLOW

1) WHEN USER AUTHENTICATES ON FACEBOOK, IT'S REDIRECTED TO THE APPLICATION MAIN PAGE WITH AN AUTHENTICATION TOKEN APPENDED ON THE URL

2) THE CLIENT SETS ITS FACEBOOK SESSION ID AND STATE ON HTML5 LOCAL STORAGE

3) THE CLIENT SENDS A GET REQUEST TO FACEBOOK OPEN GRAPH IN ORDER TO RETRIEVE ITS USER ID, NAME AND OTHER PERSONAL INFORMATION

4) ONCE THE CLIENT RECEIVES THE REQUESTED DATA, THE CLIENT WILL SAVE THIS DATA ON HTML5 LOCAL STORAGE

5) WITH THIS FACEBOOK DATA IN MEMORY, THE CLIENT WILL THEN POST A REQUEST TO MTVN USER MANAGER CHAT PROXY WITH ITS FACEBOOK USER ID.

6) THE MTVN USER MANAGER PROXY WILL CHECK THE REQUESTED FACEBOOK USER ID AGAINST FACEBOOK IN ORDER TO CHECK IF THE USER IS A VALID ONE.

7) THE MTVN USER MANAGER PROXY WILL ATTEMPT TO REGISTER THE USER WITH THE JABBER SERVER, USING THE FACEBOOK UID AS THE JABBER USERNAME.

8) THE USERJABBER PASSWORD WILL BE GENERATED WITH A COMBINATION OF THE USER ID WITH A SECRET HASH KEY, THAT WILL BE STORED ON THE SERVER.

9) THE SERVER SENDS BACK TO THE CLIENT, ITS JABBER USERNAME AND PASSWORD

USER MANAGER PROXY

ALTERNATE FLOW

1) WHEN USER AUTHENTICATES ON FACEBOOK, THE BROWSER IS REDIRECTED TO THE USER MANAGER PROXY URL THAT IS SETUP ON THE SPECIFIED FACEBOOK APPLICATION

2) THE USER MANAGER PROXY WILL GET OTHER INFORMATION FROM THE FACEBOOK USER AND WILL GENERATE THE JABBER USERNAME AND PASSWORD.

3) THE JABBER USERNAME WILL BE THE FACEBOOK UID AND THE PASSWORD WILL BE A COMBINATION OF THE FACEBOOK USER UID WITH A SECRET HASH KEY

4) THE SERVER SENDS BACK TO THE CLIENT, ITS JABBER USERNAME AND PASSWORD

USER MANAGER PROXY

SECURITY

1) THERE ARE SECURITY IMPLICATIONS RELATED TO SENDING THE JABBER USERNAME/PASSWORD IN A PLAIN TEXT (WITHOUT SSL OR ANY OTHER ENCRYPTION). THIS IS BEEN DONE TWICE IN THE AUTHENTICATION/REGISTRATION PROCESS: WHEN THE CLIENT RECEIVES THE JABBER USERNAME/PASS FROM THE USER MANAGEMENT PROXY AND WHEN THE CLIENT TRIES TO CONNECT TO THE JABBER SERVER THOROUGH HTTP-BIND (BOSH)

2) ONE POSSIBLE SOLUTION IS THE USE OF XMPP SESSION ATTACHMENT

PRIVATE ONE-TO-ONE CHAT OUTSIDE ROOM

1) IT'S NOT POSSIBLE TO MAINTAIN A PRIVATE CONVERSATION WITH A USER IN A CERTAIN ROOM IF ONE OF THE USERS LEAVES THE ROOM

2 SOLUTIONS

- ONE SOLUTION IS TO AUTOMATICALLY SUBSCRIBE THE USER TO AN 'INVISIBLE' ROOM ONCE THE USER IS GIVEN A JID FROM THE SERVER (EITHER ANONYMOUS OR AUTHENTICATED)

- ONE PROBLEM WITH THIS APPROACH IS THE ROSTER NUMBER FOR THIS INVISIBLE ROOM, WHICH COULD NEGATIVELY AFFECT THE CLIENT'S BANDWIDTH AND THE SERVER'S PERFORMANCE.

- ANOTHER SOLUTION IS TO NOT DISCONNECT THE USER FROM THE ROOM ONCE THE USER 'LEAVES' THE ROOM IN THE UI (WHEN IT CLICKS 'BACK' TO RETURN TO THE ROOM'S LIST.

- ONE PROBLEM WITH THIS APPROACH IS THE BANDWIDTH THAT WILL TAKE TO MAINTAIN ALL THE ROOMS' MESSAGE ROUTING AND ROSTER MANAGEMENT, WHICH COULD NEGATIVELY AFFECT THE CLIENT SIDE AND IN A SMALLER MEASURE THE SERVER SIDE.

ARC'S ROOMS MANAGEMENT PROXY

1) RECEIVES A GET REQUEST FROM THE MTVN CHAT APPLICATION UNIQUE ID AND RETURNS A JSON LIST OF ROOMS, INCLUDING THE CELEBRITY ROOM AND ITS CONFIGURATION IN A JSON FORMAT

2) THE CELEBRITY ROOM WILL BE BASED ON THE PUBLISHER-SUBSCRIBER FORMAT. ITS CONFIGURATION SETTINGS SHOULD INCLUDE THE NODE ID AND THE START-END TIME FOR THE EVENT.

3) THE RETURNED JSON OBJECT SHOULD ALSO INCLUDE THE NUMBER OF PARTICIPANTS PRESENT IN EACH ROOM. ONE WAY TO ACCOMPLISH THIS IS TO QUERY disco#info (http://xmpp.org/extensions/xep-0045.html#disco-roomitems) IN ORDER TO RETRIEVE THE CURRENT NUMBER OF OCCUPANTS IN EACH ROOM.

CLIENT DEPLOYMENT

1) THE CLIENT CODE WILL NEED TO STORE THE FACEBOOK APPLICATION KEY AND ITS UNIQUE ARC APPLICATION ID

2) THE CODE CAN BE DEPLOYED IN AN IFRAME AND THE CONFIGURATION SETTINGS (FACEBOOK APPLICATION KEY AND ARC UNIQUE APP ID) CAN BE APPENDED AS QUERY STRING PARAMETERS TO THE URL

3) POTENTIAL DRAWBACKS WITH THIS APPROACH ARE THE ORIENTATION ROTATION RESIZING AND THE PARENT-CHILD LIMITATIONS BET. THE IFRAME AND ITS PARENT CONTAINER, SPECIALLY REGARDING USER AUTHENTICATION IN DIFFERENT DOMAINS.

SERVER DEPLOYMENT

1) SETUP EJABBERD CLUSTER

2) SETUP BOSH (HTTP-BIND) CLUSTER: XMPP NODE JS

3) SETUP LOAD BALANCER FOR HTTP-BIND

4) SETUP MYSQL WITH EJABBERD USING AMAZON RDS

5) SETUP LOAD TEST USING TSUNG(http://vidorsolutions.blogspot.com/2010/12/load-testing-ejabberd-xmpp-server-with.html, https://pdincau.wordpress.com/2010/08/01/testing-your-xmpp-external-component-using-tsung-and-ejabberd-part-2/)

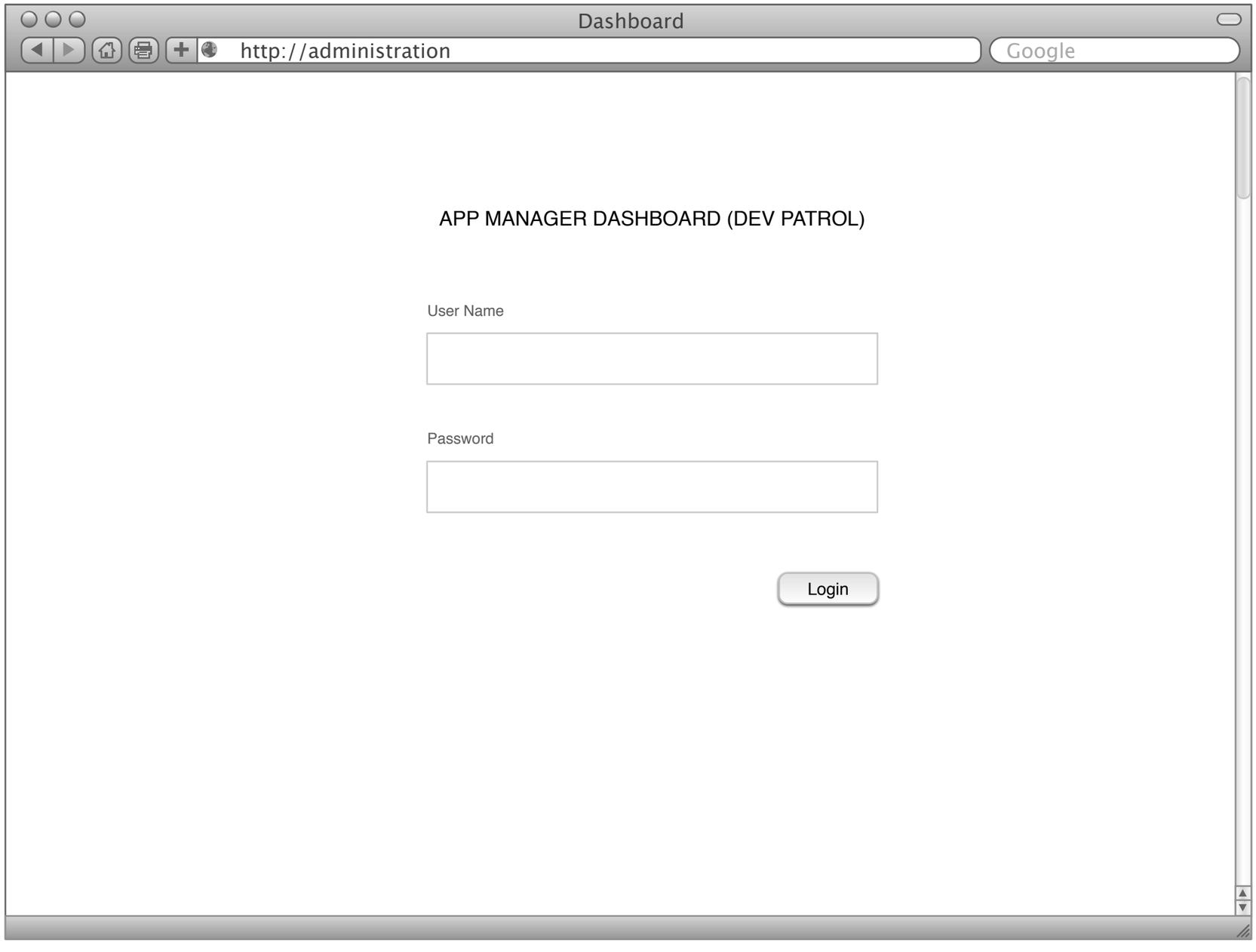
6) SETUP RIGHT SCALE INSTANCE PROVISIONING SCRIPTS

7) SETUP RIGHT SCALE TRIGGER RULES AND MONITORING

8) SETUP FINAL LOAD TESTING (XMPP AND HTTP)

*

CMS
AND CONTENT TYPES


Dashboard
http://administration
Google
APP MANAGER DASHBOARD (DEV PATROL)
User Name
Password
Login

Dashboard

http://administration

Google

Welcome User Name | Logout

My Apps | My Profile

My Apps

Actions -- Select One -- Go

New App

App	Date	Author	Published
App name	creation date	author name	published
App name	creation date	author name	unpublished
App name	creation date	author name	unpublished
App name	creation date	author name	unpublished
App name	creation date	author name	unpublished
App name	creation date	author name	unpublished

Dashboard

http://administration

Google

Welcome User Name | Logout

My Apps | My Profile

My Apps // New App

App Title

App Facebook Secret Key

App Twitter Secret Key

Cancel | Save

Dashboard

http://administration

Google

Welcome User Name | Logout

My Apps | My Profile

My Apps // App title

CHAT SETTINGS

LEADERBOARD/GAMING SETTNGS

SOCIAL MEDIA APPLICATION AND USER MANAGEMENT SETTINGS

SOCIAL MEDIA VISUALIZATION SETTINGS

Dashboard

http://administration

Google

Welcome User Name | Logout

My Apps | My Profile

My Apps // App title // CHAT SETTINGS

Actions

-- Select One --

Go

New Channel

Channel	Capacity	Type	Published	
Channel name	unlimited	celebrity	published	
Channel name	1000	regular	published	send announcement
Channel name	100	regular	published	send announcement
Channel name	500	regular	published	send announcement
Channel name	500	regular	unpublished	send announcement
Channel name	500	regular	unpublished	send announcement

Dashboard

http://administration

Google

Welcome User Name | Logout

My Apps | My Profile

My Apps // App title // CHAT SETTINGS // New Channel type Regular

Channel Name

Brief description

Thumbnail Picture Upload

Capacity

App ID/Name

Dashboard

http://administration

Google

Welcome User Name | Logout

My Apps | My Profile

My Apps // App title // CHAT SETTINGS // New Channel type Celebrity

Event/Channel Name

Date Begin

Date End

Brief description

Thumbnail Picture Upload

App ID/Name

to be continued...