

Thesis Writing Fall 2008

Experiment/Research Brief: Core Version

21 September 2008

Drew Cogbill

Writing Instructor: Chris Prentice

Thesis Instruction: Colleen Macklin

## **1. Current Thesis Concept**

Pigeon is an attempt to use accessible communication technology to provide a more affordable and accessible communication conduit for people with limited technological access in developing nations. Furthermore, Pigeon explores how web services can be delivered via different means. By demanding that all primary interaction on the system be via phone lines, Pigeon attempts to give the functionality of a social network, which are almost entirely associated with traditional web interaction, to people who may have no or limited access to computers or even complex, browser-enabled mobile phones.

## **2. Experiment/research objective**

Last week's remapping of the Pigeon, reminded me how quickly the system could be technically implemented. With even a rough and buggy version, Pigeon could be evaluated. In that pursuit, this week I planned to technically implement the ability to sign up for Pigeon and to add contacts. The previously existing prototype of Pigeon provided a technical backbone to quickly generate missing features such as signing up or the ability to add contacts. Voxeo provides the perfect prototyping platform for voice-based applications. They have a company specific markup language called CallXML, which is similar to VoiceXML. CallXML can be dynamically generated using PHP, and member information can be stored in a MySQL database. Though Voxeo may or may not be affordable and ideal for the ultimate implementation, quickly generating an alpha-working version of Pigeon will provide the opportunity to test the system showing how Pigeon may be crafted into the most effective communication tool for the target audience.

## **3. Design questions**

Most of this week's design question directly stem from the two implementation goals. For the sign up feature of Pigeon, some people could possibly enter a member number that already exists. The code must be prepared to deal with this issue. If a new member number is chosen, exactly what information will be collected at sign up must be considered. All interaction on Pigeon must be designed to minimize the amount of time a user is on the system so to minimize cost.

After signing up, the next most important feature on Pigeon is the ability to add contact with whom you can share your messages. What is the best way to add these contacts? Last weeks system map showed how if a contact request exists a user is prompted for a response at login. Exactly where in the technical login process these requests are generated must be figured out. Furthermore, once a request is accepted or declined, how should the choice be technically processed? Within all of this code, PHP is being used to generate CallXML, and logical operations that must be carried out for Pigeon can be done in either language. Though both will work, it may be good for consistency to decide how to carry out such operations.

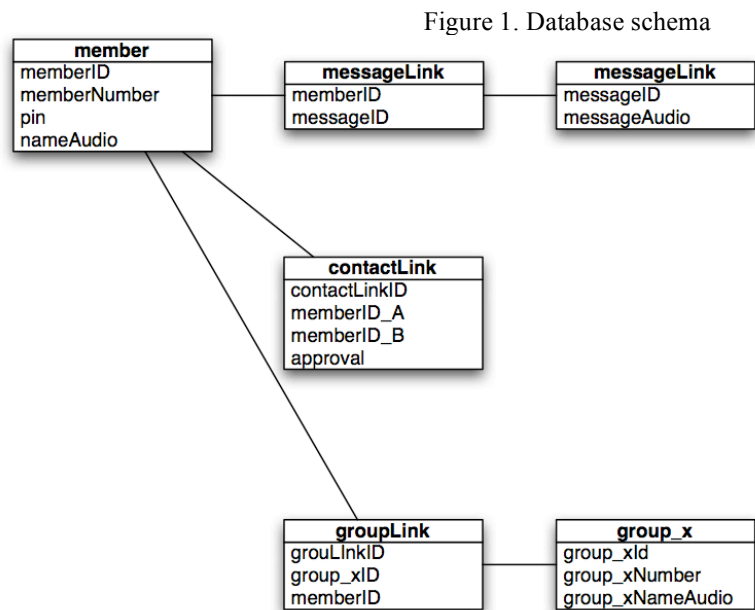
#### 4. Experiment/research description

Two prototypes were created to work out the two implementation objectives of the week. The first made it possible to sign up for Pigeon, and the second, which extended the first, dealt with allowing existing users to add

contacts. This sign up prototype is available via Skype at +99000936 9996079834 or via a direct line at (845) 521-7421. This prototype was generally easy to implement technically. Figure 1, which was created for the Pigeon prototype developed last year, shows the proposed database schema for Pigeon. When signing in to Pigeon, PHP checks to see if a given

memberNumber exists, and if so, if a given PIN matches that memberNumber. To make it possible to sign up for Pigeon, a small amount of CallXML was written to take a desired member number and PIN, and a small amount of PHP was written to insert that information into the database.

After signing up for Pigeon, another basic function of the system is the ability to add contacts. The code for implementing this feature was more complicated than the sign up code.



The contact add prototype can be accessed via direct US local (713) 568-7265 or via Skype +99000936 9996079900. There are two parts to the contact adding function. First, the ability to request a contact was created. Second, it was made possible to accept or decline a contact request from another member. The contactLink table, in the database structure, contains the information about member numbers that are connected. The approval column contains a “0” if a contact request is not currently approved, and a “1” if the request is approved.

In the contacts menu, Pigeon now allows a user to request a contact by entering their 9-digit member number. When the requested user next logs onto Pigeon, he or she is asked if they would like to confirm or decline the contact request. In the near future, when a contact is accepted, both users’ public message is available to the other, and they are able to send direct messages to one another.

## **5. Results and Conclusion**

Pigeon can now sign up new users and allow users to become contacts with other users. If a user attempts to sign up with an existing user number, a PHP script recognizes that the number exists in the member table and instead of generating a CallXML page to alert the new user that their member number and pin have been set, asks for a new number and then processes that number through the PHP script again. The logical operations of PHP are stronger than those of CallXML, so those of PHP will be used throughout Pigeon.

Only one way of how users might add contacts was created. It may be beneficial to prototype other options. The code for checking if a contact request exists was placed immediately after the code checking if the member number and PIN match. The code was not placed under the contact menu but can easily be added there. The current code can go through multiple contact requests if each is either accepted or declined but not ignored. This feature needs to be added in the future. When a request is accepted or declined, a different script for each option was created which routes the attention back to the main request checker script, which runs until each request has been handled.

## **6. Next steps**

The next implementation module that must be handled is that of message exchange. When two members are contacts, they will be able to listen to each other’s public messages. As

soon as this is implemented, user testing of the system can commence to raise and address large design questions. Certain things such as if member numbers and PINs should be entered more than once at sign up to ensure accuracy, will be easier to test with real users. Every moment of time that can be saved on the phone should be. Also with saved member information, there is currently no way of accessing lost member numbers or PINs. It should be considered, at some point, how to give this information to users.

The current method of creating contacts may or may not be the most effective, as users would have had to communicate their member numbers via another communication means outside of Pigeon. Should some type of searchable profile exist? If such a thing existed, when should this information be collected? Interaction with Pigeon must be as brief as possible to get to content, for cost saving means.

The effectiveness and brevity of the script should be tested, for the same reason. When a user logs in for the first time, the system needs to be intuitive enough for no instructions to be necessary. Beyond basic usage, there could be a multitude of hidden shortcuts for navigating through Pigeon. The best way to distribute information about navigation needs to be investigated. Even before navigation, the best way to describe the system to non-users when calling in can be constantly evaluated. Options for these questions will be able to be tested soon with a buggy, but working version of Pigeon.

## **7. Bibliography**

Davis, Michele E. and Jon A. Phillips. *Learning PHP & MySQL*. Sebastopol, CA: O'Reiley Media Inc., 2007.

“Recent immigrants driving advanced mobile phone use, both in Europe and in the US.” putting people first, 4 May 2008, <http://www.experientia.com/blog/recent-immigrants-driving-advanced-mobile-phone-use-both-in-europe-and-in-the-us/>.

Reliance Global Call. <http://www.relianceindiacall.com/web/globalhome.aspx>

TextMarks. <http://www.textmarks.com/>.

Voxeo: VoiceXML and CCXML Developer Site. <http://evolution.voxeo.com/>.