

# SmilieFace: Evolving Interface Design

*Hengky*

Department of Computing  
Curtin University  
Perth, Western Australia  
hengky@postgrad.curtin.edu.au

*Andrew Marriott*

Department of Computing  
Curtin University  
Perth, Western Australia  
raytrace@cs.curtin.edu.au

*David McMeekin*

Department of Computing  
Curtin University  
Perth, Western Australia  
D.McMeekin@curtin.edu.au

**Abstract**—Smiliemail is an existing web-based application that enhances online communication by enabling users to create and send affective, engaging messages. It uses a computer generated video avatar, which articulates a sender's message that has been tagged using an XML markup language.

With the advent of smart phones, a new less complex screen by screen interface was required due to their reduced screen size and power. A new Smiliemail client could also cater for the mobile paradigm of "media-rich communication everywhere".

Interaction between users on Social Network Sites (SNS), such as Facebook often use emotional or affective communication. However, plain text messages, even if tempered with appropriate smile icons, often cause misunderstanding, leading to flame wars or similar. Hence the Smiliemail affective avatars may provide an effective inter-user communication on SNSs.

This case-study discusses the user interface evolution from the existing web-based Smiliemail interface, through to a smart phone interface, and finally to the SmilieFace Facebook application "canvas" interface. This study also details some Facebook application user interface and implementation issues.

**Keywords**—*smilieface, vhtml, affective video, social media*

## I. INTRODUCTION

Krupansky [1] proposed that an Embodied Conversational Agent (ECA), which is a computer-generated character that demonstrates many of the same properties as humans in face-to-face conversation, may serve as a suitable avatar in online communication. An ECA can be created as a synthetic photorealistic face or as a simple cartoon character (Figure 1).



Figure 1: Photorealistic and iconic avatars

Picard [2] argued that "affective" computing will improve communication between humans and computers, and by extension, between humans if the computer is relaying messages. An "affective" message is one capable of emulating various emotions, for example by changes in an ECA's facial expression (Figure 2) as well as emotional changes in the intonation of its voice.

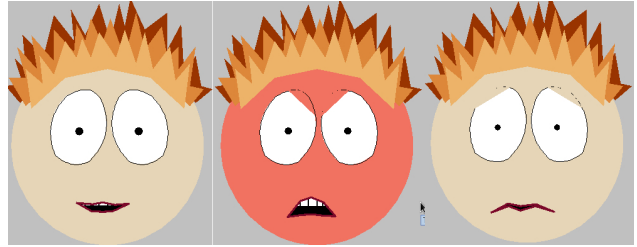


Figure 2: Emotional avatar faces

Recently there has been considerable interest in the need to better represent users in graphical environments, both in academia and in commercial applications [3]. In such environments, users are often represented as avatars [4]. Donath [5] indicated that:

*"The goal of many systems is to bring the expressive qualities of the face to a virtual world; the challenge is sensing and producing expression in a socially meaningful way"*

The Smiliemail research project ([www.smiliemail.org](http://www.smiliemail.org)) investigated the design, implementation, evaluation and use of synthetic, cartoon-like, embodied, scripted avatars as an alternative method by which affective content could be asynchronously communicated [6]. Users were able to create expressive animated avatars that spoke the entered text, and the avatar videos were sent to others via an email containing a link where the recipient could watch the video (examples can be seen at

<http://www.smiliemail.org/examples/?movies>).

The primary objective of the original interface was to enable users to easily and effectively create tagged messages to be rendered into avatar videos. The web interface for this, although a monolithic "slab", was easy to use, effective and extensible.

Within a 13 day evaluation period, 70 users registered to use the system, with 29 evaluations submitted by participants. 79% of the participants agreed that the interface made it easy to compose a message, 93% found the tags useful in expressing themselves in the message they composed, and 75% said that the smile avatar was appealing [6]. The use of computer generated talking avatars was successful in aiding communication of affective messages. An interesting "look into the future" quote came from the qualitative evaluation of the research:

*"Very nice design, but it also looks very corporate. Try a more social networking approach, like Flickr.com (go for valid css/xhtml interface perhaps?)"*

With the advent of smart phones with their increase in screen real-estate from the earlier mobile phones, Smiliemail was further developed to cater for composing and displaying affective videos on these mobile devices. The monolithic "slab" web interface was no longer appropriate, and hence a more formal user interface design approach was employed in the re-development.

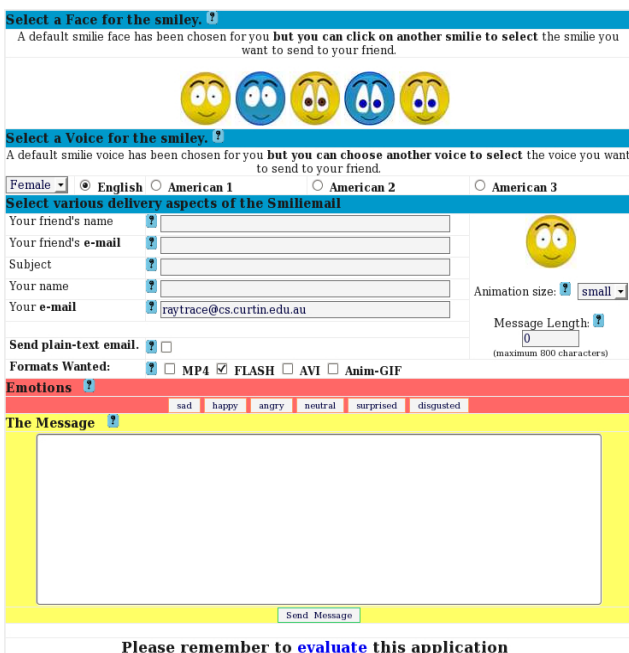
Similarly, with the advent of SNSs, a need emerged to further develop the interface to cater for a new SNS "look and feel". Facebook, Myspace, Google and others were keen to attract 3rd party applications to their sites so as to enrich the user experience.

This meant that the original Smiliemail interface needed to evolve to address these two needs.

## II. THE EVOLVING USER INTERFACE

### A. The Smiliemail Web Interface

The original Smiliemail web interface was a single monolithic "slab", iteratively developed, but built on the dictum that a naive user should be able to do things quickly and easily, whereas an expert user should also be able to do complex things quickly and easily. The naive user interface is shown in Figure 3.



Please remember to **evaluate** this application  
Figure 3: Naive Smiliemail web interface

The web interface enabled a user to choose a face, a voice type and gender, indicate the various email addresses and names, choose the size of the video created and its format, and provided a text area where the user could create and tag the message. This "naive" user interface had a limited choice of faces, video size, message length, and reduced functionality in terms of available tags. Help in using the interface was provided by tooltips that appeared when the user's cursor hovered over a component.

Examples of the various Smiliemail faces, voices, and backgrounds, as well as the audio and movies produced can be found at <http://www.smiliemail.org/examples/>. Note: the videos are small in size because of bandwidth limitations for a user to retrieve a video. Either quality or video size was sacrificed in order to reduce the file size of

the generated video. Imagine streaming the larger of the video examples - 22 megabytes - to a user in their home.

The Smiliemail systems Administrator could make a user an "expert": a user's account could be granted specific extra privileges, with an "expert" having all privileges. Figure 4 shows the expert interface, and this allowed for more faces, arbitrary resolution videos and message length, and increased control over the affective nature of the message via extra tags.



Figure 4: Expert User web Interface

Figure 5 shows a marked-up message example. The user types in the text, highlights parts of it, and then uses the emotion / action buttons to put appropriate tags around that text. The user can then modify the tags. For example, the length of the **pause** could be adjusted.

The tags use VHML - the Virtual Human Markup Language ([7], [8], [9]). This was the most appropriate ML at the time, although a long term goal is to migrate to newer standards such as emotionML [10] and BML [11].

```

<happy>
  Happy Birthday to <pause length="short"/> you
  <pause length="long"/>.
  Happy Birthday to you <pause length="long"/>.
  Happy Birthday Favourite German
  <pause length="long"/>
  <pause length="long"/>
  <pause length="long"/>.
  Happy Birthday to yooooooooooooooooooooo.
</happy>

```

Figure 5: VHML marked up text

With a scrolling web-page, this open ended design was effective and functional, and in keeping with many web-based applications of the time: a page with different functionality based upon the user's experience was dynamically presented to them. Little attention was paid to efficient layout, web page data size, or to client-server page refresh transactions. It was a time of WIMPs, fast desktops, big screens, and high bandwidth.

The advent of smart mobile devices changed that outlook- it was time for an <http://m.smiliemail.org/> server and also a planned design to specifically cater for the screen real-estate restrictions and touch screen interaction of the mobile web browser or mobile client.

### B. Mobile Devices

Kaasinen [12] introduced a "Technology Acceptance Model" wherein user acceptance of mobile phone services was based on four factors: value of service, trust, ease of adoption, and ease of use. The service will be "valuable" if it is built on the basis of utility, communication or fun. To be trustworthy, user privacy and personal data should be protected. To satisfy the ease of adoption factor, the service needs to be easy to locate, easy to use, and easy to dismiss when no longer needed. Ease of use can be achieved by enabling effortless and straight forward user interaction with the service.

These factors were seen as good goals to aim for in any planned development of a mobile interface to the Smiliemail system.

Having a concrete representation of the user interface reduces the mental burden on interface designers ([13], [14] [15]). Therefore, to create an acceptable mobile client interface to Smiliemail, a series of hand drawn screen views were developed. These would represent the user interaction on the new Smiliemail client mobile platform in a screen by screen, or View by View manner (Figure 6).

Several designs were prototyped, and the views were then coded (Figure 7). Although not used in this research, related systems used GUI builders to create fast UI prototypes for analysis and testing from the pencil and paper designs of the views.

Each View led forward or backward to another View so users could compose or view a Smiliemail video. The monolithic, scrolling "slab" interface was replaced by the typical mobile device paradigm of separate Views and touch screens. Similarly, the mobile platform's GUI components were used instead of a web Form.

The pencil and paper designs, coupled with the rapid prototyping, produced more effective and usable interfaces in a shorter time than by straight coding of the interfaces. Application interface-flow issues were uncovered at an earlier stage as well.

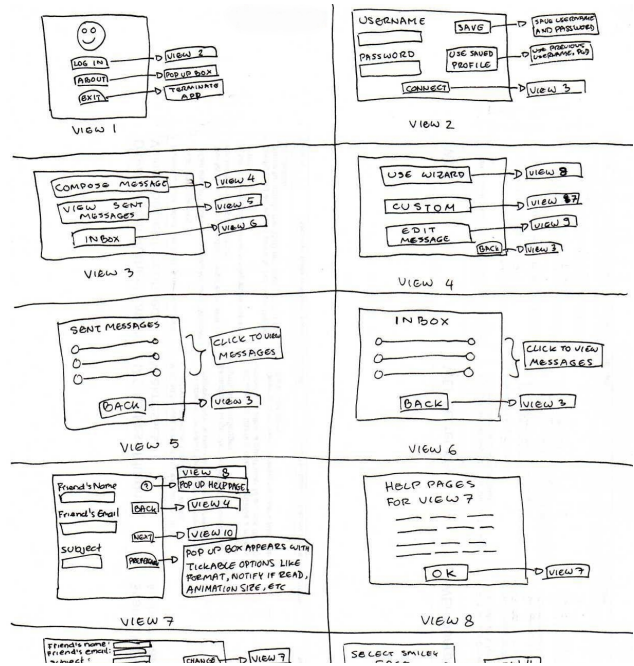


Figure 6: Mobile client design



Figure 7: Mobile Client Views

The Smiliemail server was also modified so that the client was able to read and write information from and to the database that held the user and message information. This was done via standard http requests to the server.

Using Kaasinen's "Technology Acceptance Model", the new mobile client was evaluated against the model's acceptance factors: value of service, trust, ease of adoption, and ease of use.

The developed client was tested using existing messages, and hence "value of service" was not tested, although the service was utilitarian, communicative and fun.

The client adopted the mobile service "trust" paradigm of storing the user details on the mobile device. It may have been possible for the device to be used by more than one person and hence security of information was addressed as an issue when considering password retrieval.

"Ease of adoption" was not an issue, as the underlying Smiliemail system already existed and provided a useful service. The mobile Smiliemail interface addition was well planned and executed. One potential "adoption" issue identified was the video formats supported on specific platforms. This was solved for the Android platform and catered for on the iPhone platform, but the introduction of newer video technology may cause users to be wary of any application that can't display any new video format [16].

Since the client followed mobile platform interface design guidelines, was pencil and paper designed and then prototyped before being implemented, the "ease of use" factor, although not formally tested, was considered good by typical mobile phone application users. It looked like any other mobile "app".

The ability for an expert user to do things quickly was catered for but in a reduced manner, due to the View by View navigation overhead, and the reduced screen real-estate. GUI component "Help" tooltips were replaced with a View specific help button on each View page.

This successful mobile interface development of Smiliemail was motivated by the mobile devices' platform improvement: faster cpu, higher screen resolutions, and better base-level graphical components than their predecessors. And of course, the escalating adoption rate of mobile technology by users.

Mobile clients were facing another development - that of Social Network Sites, and of accessing a user's "space".

### C. Social Network Sites

Because of their proliferation, SNSs were growing in importance as communication centres. Boyd and Ellison [17] defined SNSs as:

*web-based services that allow individuals to:*

1. *construct a public or semi-public profile within a bounded system,*
2. *articulate a list of other users with whom they share a connection, and*
3. *view and traverse their list of connections and those made by others within the system. The nature and nomenclature of these connections may vary from site to site.*

Item 1 above suggested that on SNSs, users may wish, for privacy or other reasons, to create a digital persona/avatar so as to seek out or contact new "friends".

Golder, Wilkinson, and Huberman [18] analysed 362 million messages exchanged by 4.2 million Facebook users during a 26 month interval and concluded that whilst users tended to exchange messages mostly with their friends, the number of message partners with whom they regularly exchanged messages was fairly low. Facebook provides 3 mechanisms for user communication:

1. **Wall Post:** a message written onto the recipient's profile page and is visible to everybody.
2. **Private Message:** an email like message only visible to its recipient.
3. **Poke:** a content-less message and is usually used to get attention from its recipient.

Interaction between users in an SNS is a key factor that influences its popularity, as the sites have become a popular way for users to keep in touch, express themselves, and share content. Donath and Boyd [19] and Viswanath et al [20] examined user interactions and learned that 'Wall' posts, are the most popular user interaction method. The number of 'Wall' posts between two users usually reflects the strength of the ties they share: sporadic or close to no 'Wall' post exchanges usually signifies that the ties between the users are quite weak, whilst regular or intense 'Wall' posts usually indicates that they share much stronger ties.

### D. SNS applications

Gjoka et al. [21] proposed that 3rd party applications were a key feature that contributed to the unprecedented success of Facebook. They classified the seven most popular categories or types of application in Facebook as:

- **Friend comparison:** allows users to declare best friends and compare friend traits.
- **Casual communication:** allows users to exchange messages and write on each others wall.
- **Rating, Recommendations, Taste Matching:** enables users to review, compare and recommend items spanning from music to restaurants.
- **Gestures:** allows users to perform virtual gestures.
- **Self Expression:** enables users to express moods, political opinions, etc.
- **Gifting:** allows users to exchange virtual gifts.
- **Meeting People:** allows users to find people of similar interests.

Gjoka et al. [21] collected data over a 170 day period of the 100 most active applications, and of these, the "Friend comparison" and "Casual communication" types proved to be the most popular application type. This indicated that Facebook users primarily used it as a means to declare how they feel as well as a means to communicate by exchanging messages. **Gifting** was another popular application type.

In addition, Lange's [22] research on YouTube usage suggested that sending videos to friends deepened those "friend" connections.

These factors implied that an affective messaging system that allowed users to send each other video messages would be beneficial. A new interface to the Smiliemail system was designed and implemented so as to provide affective videos via a 3rd party application - SmilieFace.

E. The SmilieFace Application

Facebook provides a webpage with a "canvas" area where a 3rd party application can display its interface, similar to the original Smiliemail interface mechanism. However, with the redevelopment of the original interface, the UI was again properly designed via pencil and paper (partly shown in Figure 8). The driving basis for this development was the need to conform to the underlying formal Facebook application's "look and feel". It needed to use the Facebook "canvas" - web based but restricted in size - and also needed to cater for mobile device access.

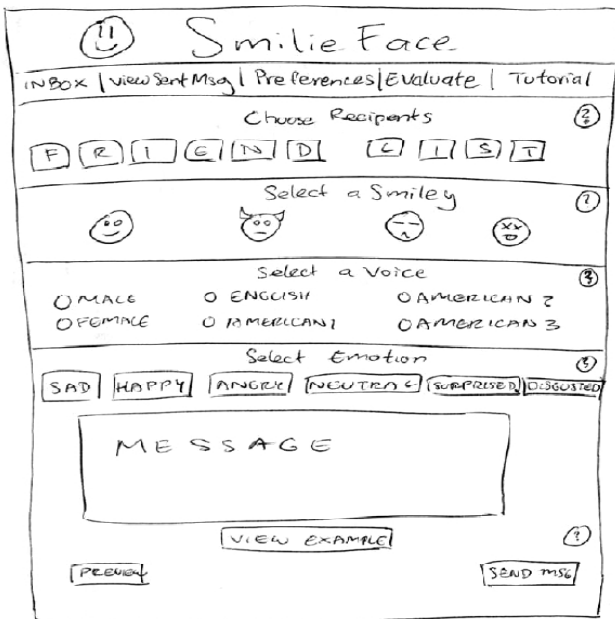
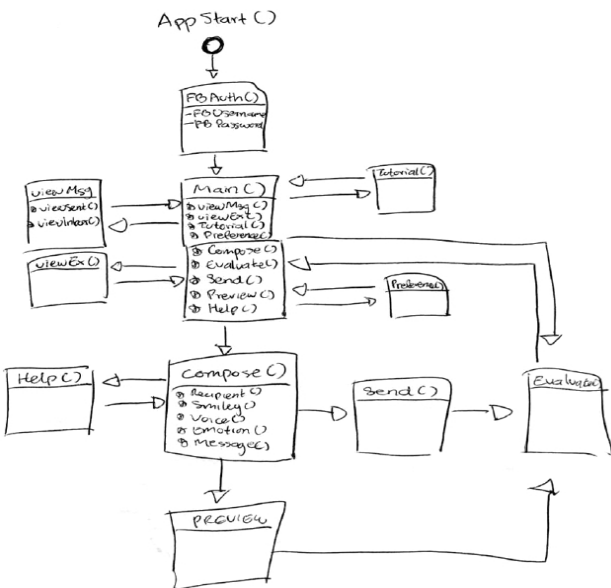


Figure 8: SmilieFace pencil design

The complexity of the system and the user interaction also required that the Unified Modeling Language (UML) [23] be used to ensure correct user interaction in the implementation (partly shown in Figure 9).



UML Diagram Design A

Figure 9: UML for SmilieFace application

The implemented view is once again a monolithic "slab" (Figure 10). Parts of the image are blurred to preserve the privacy of this Facebook user. As Figure 10 shows, the design is similar to most email compose pages and should be intuitive and easy to use for the users.



Figure 10: SmilieFace application canvas

Due to the newness of the Facebook application canvas technology, prototypes were not developed, but an iterative design and implementation approach that followed the pencil and paper design, and conformed to the UML use-case scenario produced an effective application.

The use of "tabs" provides the user with a smoother web browsing experience whilst navigating through the pages, as the browser would not have to reload a page for a feature or preferences request. The SmilieFace Interface is also designed to be clutter free to cater for users that accessed the application through devices with small screens such as smart phones, tablet PCs, etc (although not tested in this study).

The re-use of the "slab" is not really surprising as Mutale [6] showed that the initial web interface of Smiliemail was effective and easy to use. The original SmilieFace interface has been updated to reflect the need for the application to conform to Facebook's "look and feel", and to be mobile client ready. The Smiliemail server was also updated to communicate with the application.

The SmilieFace system is due for evaluation in early 2012, with similar questions to the original study such as ease of use, but also with aspects that deal with the Social Media nature of the research.

#### F. Facebook Implementation and Interface Issues

During the implementation of this interface, Facebook decided that their canvas functionality needed improving (<http://developers.facebook.com/blog/post/538/>).

The interface canvas for our Facebook application had been built using Facebook Markup Language and Facebook JavaScript as the primary technologies, but these have recently been deprecated and the Software Development Kits (SDK) extensively revised (<http://developers.facebook.com/blog/post/479/>).

New PHP and Javascript SDKs have been released and all 3rd party applications are required to migrate to them for creating, implementing and managing the interfaces (<http://developers.facebook.com/blog/post/525/> and <http://developers.facebook.com/blog/post/503/>).

And in order to provide a more secure user experience, Facebook also introduced a new method for authentication (<http://developers.facebook.com/blog/post/497/>).

These changes, whilst welcome, came at a bad time, causing delays and re-development in the SmilieFace user interface. Facebook hopes that these changes will improve the overall user experience whilst navigating through any application interface.

There are many ways for a user to access Facebook, (smart phones, tablet, PC), and the design of SmilieFace has taken into consideration that the application might be accessed by various web browsers (IE, FF, Chrome, Safari) with different versions. Unfortunately, different web browsers have different "standards" implementations, and some compatibility issues have arisen whilst viewing SmilieFace in a certain web browsers. The most outstanding problem is that there is an issue that affects the viewing of SmilieFace in Internet Explorer 9 (IE9), where instead of displaying parts of the web page in different tabs, IE9 appends all the different parts and displays them together in the same page. There is an official fix for this "bug" that we implemented on the SmilieFace server, but this header information fix is not passed through to the browser by the Facebook "canvas" displaying code. The header information is sent to the browser from the page enclosing the canvas, and the canvas header from our server, with its fix, is discarded.

Statistics released by Pachal and StatCounter show that Chrome 15 (24.55%) tops the worldwide browser market, edging out IE8 (22.16%) and FireFox (15.53%). However, by taking all the versions of the web browsers under considerations, IE remained the most popular web browser (38.5%), with Chrome second (27.08%) followed by FireFox (25.55%). From these statistics, we reasoned, that in the short term, the IE9 compatibility issue would not affect that many users since IE8 is the most popular version of the web browsers. This issue needs to be addressed by Facebook's SDKs in the future.

There are also format compatibility issues with displaying the SmilieFace videos, especially when catering for mobile platforms, notably the lack of a Flash player on iPads, but these can be overcome by generating two versions of the video and streaming the appropriate one. This adds to the production time, but is not onerous.

Informing users of incoming videos was also another User Interface challenge as an application has to comply with Facebook's policy in protecting the privacy of users.

A "Wall post" is visible to all friends, and this may violate the senders' and recipients' privacy although it is fully under the user's control. An "application request" is only to the recipient, but lacks the personalization and visibility of the "wall post". User will have full control on which of the two is preferable to them, however the default setting will be an "Application Request". This takes the users' privacy into consideration.

One application issue is that if SmilieFace wants to become a successful and popular 3rd party application, it must facilitate a way for its users to refer the SmilieFace Application to their friends. Sharing created videos may encourage the adoption of the application, but may also infringe the original recipient's privacy, similar to forwarding email without consent. And overexposing the application can be a violation of Facebook's "Don't Spam" policy or could cause annoyance to the potential users causing them to block the application without even using it. Hence, certain server limitations have been put in place both to protect the users' privacy as well as conforming to Facebook's strict policies. This limitation involved limiting the number of users that can receive a SmilieFace video message at one time, hence limiting the number of potential "Wall Post" and "Application Request" sent out at the same time.

One final issue that may occur is that the Smiliemail server is an 8 core PC, with 24 gigabytes of main memory communicating to the world via an ADSL2+ network connection. This may sound like a powerful server, but what would happen if the SmilieFace application became very popular? What if 100 users installed it and used it? What if 1,000 or 10,000 users installed it? Facebook encourages this as they can make money from putting advertisements around the application's canvas. But if a video takes 1 minute to produce and is 1 megabyte in size, then 10,000 users would easily swamp this server and its network connection. Response times would drop and the application would not be evaluated as useful nor usable. The researchers want SmilieFace to be popular, but not too popular. A production system would use cloud computing resources that could scale with demand. Facebook has recently partnered with "Heroku" to make it even easier to deploy Facebook applications ([24]). Heroku is a cloud service provider that is able to host 3rd party Facebook applications.

Although not an obvious user interface issue, this scalability issue does affect the user experience in using the application, and hence needs to be catered for in the overall application design. Scalability becomes a new factor in Kaasinen's 2005 "Technology Acceptance Model" when dealing with SNSs applications.

### III. CONCLUSIONS

The original Smiliemail system was successful in meeting its objectives. The developed mobile Smiliemail client, although not formally evaluated, was also successful when measured against a Technology Acceptance Model. The development of this client was made easier by adhering to established mobile platform interface design guidelines, and by using a planned pencil and paper design followed by prototyping before coding.

The SmilieFace canvas application development also followed strict Facebook design guidelines, was pencil and

paper designed, and then iteratively developed and implemented. Its integration with and communication to the Smiliemail system is ready for testing and evaluation, and is scheduled for early 2012.

The SmilieFace application is almost an instantiation of the initial Smiliemail system evaluation feedback: "Very nice design, but it also looks very corporate. Try a more social networking approach,...". Smiliemail has evolved to stay up to date with today's technology and way of life.

New platforms arrive, new paradigms are used, more screen space, less screen space, faster, slower, different requirements. From web-based email for videos, to Facebook embedded video applications, the intent is the same - effective affective communication. One of the results of this study will be an analysis of the acceptance and effectiveness of a Facebook application for composing and displaying affective avatar-based video messages.

#### REFERENCES

- [1] J. Krupansky, "Definition: Embodied Conversational Agent (ECA)," 23-Sep-2010. [Online]. Available: [http://activity.com/def/embodied\\_conversational\\_agent.htm](http://activity.com/def/embodied_conversational_agent.htm).
- [2] R. Picard, *Affective Computing*. USA: M.I.T Press, 1997.
- [3] T. Takahashi, Y. Katagiri, and C. Bartneck, "Show Me What You Mean - Expressive Media for Online Communities," in *CHI2003 Workshop on Subtle Expressivity of Characters and Robots*, 2003, vol. 18-21.
- [4] M. Gillies, "Integrating Autonomous Behavior and User Control for Believable Agents," in *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2004*, 2004.
- [5] J. Donath, "Mediated Faces," in *Proceedings of the 4th International Conference: Cognitive Technology: Instruments of Mind*, 2001, pp. 6-9.
- [6] G. L. Mutale, "Smile-Mail: An Avatar-based Asynchronous Communication Application," Curtin University of Technology, Perth, Western Australia., 2005.
- [7] C. Gustavsson, L. Strindlund, and E. Wiknertz, "Verification, Validation and Evaluation of the Virtual Human Markup Language (VHML)," Linköping University, Sweden., 2001.
- [8] A. Marriott and S. Beard, "VHML Standardisation," in *European Union IST Special workshop on VHML*, Bruxelles, Belgium, 2000.
- [9] A. Marriott and J. Stallo, "VHML- Uncertainties and Problems... A discussion," presented at the AAMAS 2002, Bologna, Italy, 2002.
- [10] P. Baggia, F. Burkhardt, C. Pelachaud, C. Peter, E. Zovato, and M. Schröder, "Emotion Markup Language (EmotionML) 1.0." [Online]. Available: <http://www.w3.org/TR/emotionml/>. [Accessed: 09-Jan-2012].
- [11] SAIBA, "BML 1.0 Standard." [Online]. Available: <http://www.mindmakers.org/projects/bml-1-0/wiki/Wiki>. [Accessed: 09-Jan-2012].
- [12] E. Kaasinen, "User acceptance of mobile services - value, ease of use, trust and ease of adoption," 2005. [Online]. Available: <http://www.springerlink.com/link.asp?id=20f78wm8rer>.
- [13] L. Cardelli, "Building user interfaces by direct manipulation," in *1st annual ACM SIGGRAPH symposium on User Interface Software*, New York, NY, USA, 1988, pp. 152 - 166.
- [14] B. Myers, S. E. Hudson, and R. Pausch, "Past, present, and future of user interface software tools," *ACM Transactions on Computer-Human Interaction*, vol. 7, no. 1, pp. 3 - 28, 2000.
- [15] H. Jones, "18 Great Examples of Sketched UI Wireframes and Mockups | Inspiration." [Online]. Available: <http://webdesignledger.com/inspiration/18-great-examples-of-sketched-ui-wireframes-and-mockups>. [Accessed: 11-Jul-2010].
- [16] S. Jobs, "Thoughts on Flash." [Online]. Available: <http://www.apple.com/hotnews/thoughts-on-flash/>. [Accessed: 10-Jan-2012].
- [17] D. M. Boyd and N. B. Ellison, "Social network sites: Definition, history, and scholarship," *Journal of Computer-Mediated Communication*, vol. 13, no. 1, pp. 210-230, Oct. 2010.
- [18] S. A. Golder, D. Wilkinson, and B. A. Huberman, "Rhythms of social interaction: messaging within a massive online network," *Proceedings of the 3rd International Conference on Communities and Technologies*, 2007.
- [19] J. Donath and D. M. Boyd, "Public Displays of Connection," *BT Technology Journal*, vol. 22, no. 4, pp. 71-82, 2004.
- [20] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, "On the evolution of user interaction in Facebook," *Proceedings of the 2nd ACM workshop on Online social networks*, 2009.
- [21] M. Gjoka, M. Sirivianos, A. Markopoulou, and X. Yang, "Poking facebook: characterization of osn applications," *Proceedings of the first workshop on Online social networks*, 2008.
- [22] P. G. Lange, "Publicly private and privately public: Social networking on YouTube," *Journal of Computer-Mediated Communication*, vol. 13, no. 1, Oct. 2010.
- [23] G. Booch, I. Jacobson, and J. Rumbaugh, "OMG Unified Modeling Language Specification," 2000. [Online]. Available: <http://www.uml.org/>. [Accessed: 10-Jan-2012].
- [24] C. Lee, "Facebook and Heroku: an even easier way to get started - Facebook developers." [Online]. Available: <https://developers.facebook.com/blog/post/558/>. [Accessed: 08-Dec-2011].