

# MuttZ: An Interactive Cellphone Game

Technical Report: CS04-26-00

Ken MacGregor

Martin Kirsten

Mouton Olivier

Department of Computer Science  
University of Cape Town

{ken, mkirsten, jolivier}@cs.uct.ac.za

## ABSTRACT

We describe the implementation of a cellphone game that generates and maintains interactivity without requiring physical player proximity or concurrency of play, and does not rely on the response of an “opponent” player to advance the game. The interactive game is implemented as a virtual pet game where players interact with each other through their virtual pets.

## 1. INTRODUCTION

Cell phones are ideal communication devices: they are compact, and allow users to communicate wherever they are. Cell phones are also an ideal device on which to play games, especially considering that cell phone users invariably carry their cell phones on their person wherever they go. The combination of these characteristics (i.e. a mobile communication device that can play games) would therefore make a cell phone the ideal device on which to implement interactive games.

Currently, the majority of interactive cell phone games are implemented using wireless technologies like Bluetooth and infrared. This allows players to link up their cell phones and to play games against each other. However, players are required to be in physical proximity (usually the same room – infrared has the additional restraint that cell phones need to be kept perfectly aligned). In order to interact, players also need to play simultaneously.

Location-based games are a recent innovation in interactive cell phone games. In these games, a player’s cellular provider locates other players in the vicinity, and the player has the ability to play against these other players, usually by launching some form of attack.<sup>1</sup> These games have removed the physical proximity required by its Bluetooth and infrared counterparts, but players still need to play concurrently in order to interact.

Another approach to interactive cell phone games are turn-based games where players send text messages to each other during their turn. Since players are able to retrieve these text messages at their convenience, and because text messages do not require users to be located near each other<sup>2</sup>, these types of interactive cell phone games require neither physical proximity nor concurrency of play. However, the game can only progress at the speed in which players respond in taking their turns, and the game state freezes while a player is waiting for his opponent’s response.

We present an implementation of a cell phone game that requires neither player proximity nor concurrent play to achieve interactivity, and does not rely on one opponent’s response in order for the game to progress. The implementation is based on the concept of a virtual pet, but this is not the only type of game that can be implemented based on our interactive game concept.

## 2. RELATED WORK

Virtual pets are creatures that inhabit a digital environment and need attention from their owner in order to grow and survive. The attention an owner can give his pet can range from feeding the pet, to letting it exercise, sleep, clean its environment, and even procreate.

Virtual pet games have proven to be very popular amongst the gaming community. The most successful commercial implementation was arguably the Tamagotchi craze of the mid nineties. Virtual pet games have also been created for the computer, from simple screensavers that display pets and do not require any input<sup>3</sup>, to more complex games like *Creatures* and *The Sims*, to Internet virtual pet sites like *Neopets*.<sup>4</sup> Virtual pet games have also been created for the cell phone. Bandai, the creators of the original Tamagotchi, created the

---

<sup>1</sup> The BotFighters game ([www.botfighters.com](http://www.botfighters.com)) in Europe and Gunslingers in Singapore (<http://www.java.com/en/connection/singtel/games/gunslingers.jsp>) are examples of location-based games.

---

<sup>2</sup> In fact, depending on agreements between cellular service providers in different countries, text messages can be sent and received anywhere in the world.

<sup>3</sup> E.g. the ubiquitous fish tank screensaver.

<sup>4</sup> [www.neopets.com](http://www.neopets.com)

first cell phone-based virtual pet game in 1997, and called it Tamapichi. Other mobile virtual pet games soon followed.

Interactive virtual pet games, on the other hand, are not as prevalent as the solo-play versions. An interactive version of the Tamagotchi game was released in Japan in March 2004, and sales have already exceeded one million.<sup>5</sup> The game was released in America on 15 August 2004. This version of the game allows virtual pets to interact by using infrared to let the Tamagotchi's play together on one device.

We chose to use a virtual pet game to implement our interactive cell phone gaming concept because of the proven popularity of this type of game, and also because it allows for an intuitive implementation of the concept.

### 3. METHOD

Our game, **MuttZ**, consists of two components: the game, which is played on a cell phone, and a server infrastructure that supports the playing of the game.

#### 3.1 The MuttZ Game

The **MuttZ** game has two gameplay “modes”: solo gameplay and interactive gameplay.

##### 3.1.1 Solo Gameplay Mode

In solo gameplay mode, the player interacts with his mutt by feeding it words, which he obtains as word packs of ten words from the **MuttZ** server. Each mutt has certain emotional attributes, and feeding it words affects the mutt's emotional disposition. Depending on the emotional slant or content of a word, the mutt's emotional disposition is affected in a certain direction: aggressive, peaceful, neat, or messy. A mutt's physical appearance changes<sup>6</sup> as its emotional disposition changes: as the attributes change, the mutt mutates to match its “feelings”. Figure 1 shows a “messy” mutt.



Figure 1: A mutated mutt

Words not only affect a mutt's emotional disposition, but also its size. The more Words a mutt is fed, the bigger it gets. A mutt also needs to be fed regularly (at least once a day) otherwise it will shrink; it dies if neglected for too long. However, feeding a mutt too often will also cause it to become sick and could result in its death. A mutt also needs to be fed regularly (at least once a day) otherwise it will shrink; it dies if neglected for too long. However, feeding a mutt too often will also cause it to become sick and could result in its death.

Once a word has been fed to a mutt, it is “used”; if the words are depleted, the player needs to download a new word pack from the server.

##### 3.1.1 Interactive Gameplay Mode

In interactive gameplay mode, different players' mutts interact with each other on their respective owners' cell phones. An interaction is initiated by one player sending an invitation to another player, inviting his mutt to interact. Included with that invitation is a host “interaction word”. This word is fed to the guest mutt if his owner accepts the invitation.<sup>7</sup> Players also have the option of including a short message to the guest mutt's owner with the invitation.

The owner of the guest mutt accepts the invitation by sending a guest interaction word to the host (as well as a message if he wishes), which is fed to the host mutt.<sup>8</sup> Both players see the effect that this feeding has on their own as well as the “opposing” player's mutt on their respective cellphones. An

<sup>5</sup> [http://www.bandai.com/news/news.cfm?wn\\_id=73](http://www.bandai.com/news/news.cfm?wn_id=73)

<sup>6</sup> Animations are achieved displaying an animation image strip frame-by-frame.

<sup>7</sup> Guest players do not know what word has been sent with the invitation unless they accept the invitation, and once an invitation has been accepted, it can't be rejected.

<sup>8</sup> Host players do not have the option to reject an acceptance of their invitation.

interaction therefore “occurs” on both cellphones, but not necessarily simultaneously. In fact, more often than not the mutt interactions will be viewed asynchronously by the players since the interaction occurs on the guest player’s cellphone as soon as he accepts the invitation and sends his interaction word to the host. The interaction will only occur on the host mutt’s cell phone when he is informed that the invitation has been accepted.

As in solo gameplay, the feeding of these words “uses up” that word, and a player who has used all his words will have to download a new word pack from the server.

An interaction completes once the effect (if any) of the two feedings has been calculated and displayed to the user (if a mutation occurs). The details of each “opposing” mutt is then deleted from the other player’s cellphone, and they can continue either interacting with other players’ mutts (i.e. continue interactive gameplay mode), or they can interact with their own mutts (i.e. play in solo gameplay mode).

One further aspect of interactive gameplay requires explanation: just as a mutt needs to be fed in order to survive, it also needs to socialize (i.e. interact) with other mutts in order to survive. If a mutt does not interact with other mutts on a regular basis, it grows sick and could die.

### 3.2 MuttZ Server Infrastructure

The server infrastructure supports and coordinates the **MuttZ** game. It consists of a server, database, WAP client and web client.

#### 3.2.1 Web Client

The **MuttZ** web-interface serves as registration page for the game, as well as a gathering place for the **MuttZ** player community. Here, players can view their own and other players’ mutts’ statistics, including pictures of all the mutts. The game itself is also downloaded to the player’s cellphone from the web-interface.

The web client also includes an Administrator Control Centre, which allows a person with administrator privilege to access the database to upload new mutt images and word packs, and to perform various other administrative functions. This ensures that the **MuttZ** environment can be updated in order to maintain player interest.

#### 3.2.2 WAP Client

The WAP client allows players to access a scaled-down version of the web client from their cell phones. Players can register to play **MuttZ** and download the game from the WAP client. Figure 2 shows the WAP client’s Welcome screen being accessed via the cell phone.



Figure 2: WAP client accessed on cell phone

#### 3.2.3 Database

The database stores all player and mutt details, including invitations that have been sent, accepted and rejected, the outcome of interactions, as well as word packs available for download.

The database is automatically updated with a mutt’s details whenever a player connects to the server if the mutt’s details have changed since the last update. This ensures that the database remains up to date.<sup>9</sup>

#### 3.2.4 Server

The server is responsible for coordinating all communications between players (sending and accepting interaction invitations), as well as to allow players to download new word packs. It also acts as middle layer to connect all the other components that make up the server-infrastructure with each other.

Players connect to the server from their cell phones via an HTTP connection. As discussed above, a mutt’s details are automatically uploaded to the database by the server every time a player makes a connection if the mutt’s details have changed since the last update was made. The server also automatically downloads certain data to a player’s cellphone when he connects. This includes details of invitations that have been sent to the player, as well as invitations that this player has sent, and that have been accepted or rejected.<sup>10</sup>

<sup>9</sup> If a player has not connected to the server for a while, the mutt details in the database could become outdated.

<sup>10</sup> If an invitation has been accepted, the interaction occurs on the host player’s cellphone as soon as he is informed of the acceptance. On the other hand, if

When required, the server needs to download images to a player's cellphone. This will occur when an interaction occurs and the opponent player's mutt mutates as a result of the interaction since the cellphone on which the interaction occurs does not necessarily have the required mutation image for the opponent's mutt.<sup>11</sup>

Due to the limited memory capacity of cellphones, not all animations can be stored in the cell phone at a time. Our solution to this restriction is to "swap out" animations as they are needed, deleting used or unnecessary animations and replacing them with ones that are likely to be used in the near future. For this purpose, an algorithm was formulated that finds the likely future animations, and these are then downloaded from the server.

## 4. RESULTS

**MuttZ** was developed on a cell phone emulator, and the solo game was ported to an actual cell phone.<sup>12</sup> However, due to limited resources we were not able to establish communication between the actual cell phone and the server. Testing of the game was therefore done on two computers linked over a local area network (LAN), with an emulator running on each. The server and database was also hosted on one of the computers.

We conducted a test of the server using test players specifically to determine whether an interactive experience was achieved, in other words whether the test users felt that they were playing "against" other players whilst playing **MuttZ** in interactive mode. In general, the test users felt that they were playing against other players, and that an interactive experience was generated by the game.

## 5. CONCLUSION

The implementation of our interactive cell phone game concept using a virtual pet game achieved in generating an interactive experience when tested in an emulated environment. Ideally the game should be tested in an actual cell phone environment on actual cell phones, but our testing delivered promising results that cell phone gaming can be interactive without requiring physical proximity or concurrency of play, and that the absence of these does not hinder the progress of the game.

## 6. ACKNOWLEDGEMENTS

We would like to thank Elsabe van Niekerk, the art student that we collaborated with to create the animations and artwork used in the game, without

---

an invitation has been rejected, the word that was originally sent with the invitation must be restored to the host player's word pack.

<sup>11</sup> The cell phone on which the interaction occurs first verifies whether in fact it does not have the appropriate image before it is downloaded.

<sup>12</sup> Nokia 6230.

whose talent and input **MuttZ** would not nearly be as aesthetically pleasing as it is.

## 7. REFERENCES

1. Optimizing the Client/Server Communication for Mobile Applications, Part 1. Forum Nokia. Version 1.0; February 28, 2003.  
<http://www.forum.nokia.com/main.html>.
2. Optimizing the Client/Server Communication for Mobile Applications, Part 2. Forum Nokia. Version 1.0; May 13, 2003.  
<http://www.forum.nokia.com/main.html>.
3. Optimizing the Client/Server Communication for Mobile Applications, Part 3. Forum Nokia. Version 1.0; October 2, 2003.  
<http://www.forum.nokia.com/main.html>.
4. Eric Giguere. Client-Server Communication Over Http Using Midp And Servlets.  
<http://java.sun.com/developer/J2METechTips/2001/tt0820.html>
5. Frédéric Kaplan. Free creatures: The role of uselessness in the design of artificial pets.  
<http://www.csl.sony.fr>
6. Multi-Player MIDP Game Programming.  
<http://www.forum.nokia.com/main.html>
7. Gary Polson. Wireless Virtual Pets for Mobile Phones: An Industry Study.  
<http://www.virtualpet.com/vp/media/mpets/mpets.htm>

This document was created with Win2PDF available at <http://www.daneprairie.com>.  
The unregistered version of Win2PDF is for evaluation or non-commercial use only.