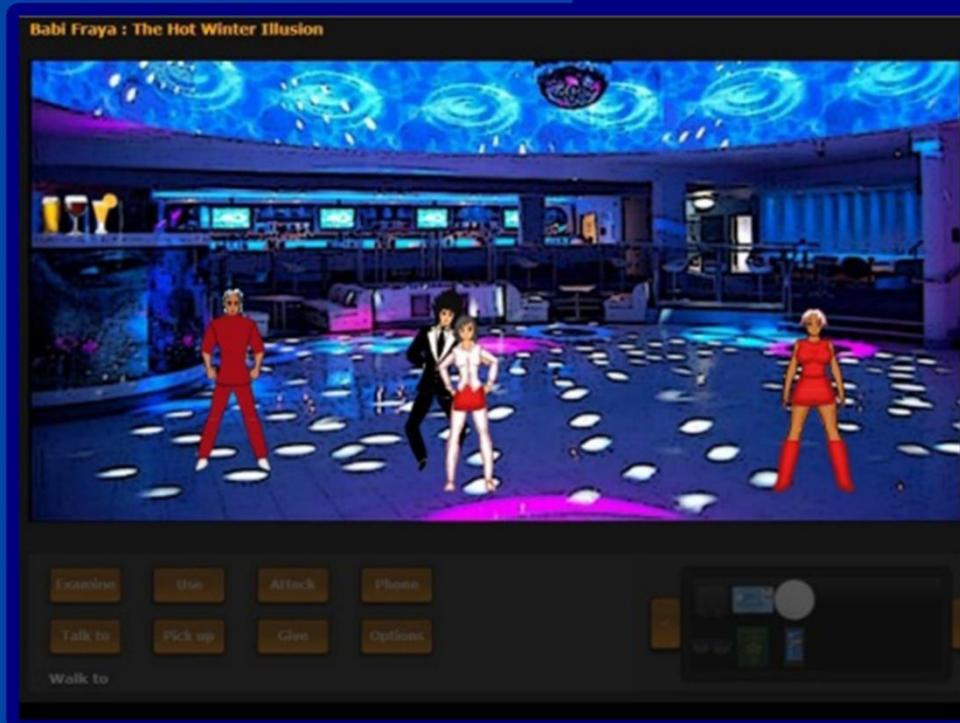


The NeceMoon Album

Technologies and Strategies to Keep Moving Forward

Chapter 8

Epic Prototypes, Classic Projects, Historic Genre



By **Necemon Yai**

The NeceMoon Album

Technologies and Strategies to Keep Moving Forward

By Necemon Yai

First edition

Published by Evasium®

April 2018

London, UK

TheAlbum.NeceMoon.com

The contents of this file are protected under the **UK Copyright, Designs and Patents Act 1988**.

License

This file is free to distribute and give away to as many people as you would like.
I only ask that you do not sell it or publish the content onto any website.
If you use any quotes from this document, give me credit and link to the original file.
If you write a book and I quote you, I will give you credit and links too.

© Necemon Yai

necemon@gmail.com

www.necemonyai.com

All Rights Reserved.

Version 1.0.7.235

To the ones I lost, to the ones I got back

For each one that gets lost, ten get laid back

Table of Contents

Introduction.....	5
For All Practical Purposes	8
Chapter 8: Epic Prototypes, Classic Projects, Historic Genre	9
Inside Bavardica - Part 1 : Discussion of the subject areas	10
Inside Bavardica - Part 2 : Progress and Achievement of the Project	13
Inside Bavardica - Part 3 : Design of the 2D characters.....	19
Inside Bavardica - Part 4 : Scope for future improvements	23
10 things I learned from building Bavardica.....	25
Babi Fraya: The Hot Winter Illusion.....	27
Relaunch : Kpakpatoya 2.0 as an Interactive Platform.....	30
Flux Project : Social News Web Applications.....	36
Stay up to date with your favorite topic on social platforms.....	38
Furtivue (Or How To Send Furtive Messages Like A Ninja)	40
Final Reminder.....	41
Conclusion	42
Share The Album with Your Mates.....	43

Introduction

About The NeceMoon Album: what is this all about?

[The NeceMoon™](#) is a Blog about Technology and Strategy. [The Album](#) is The Best-Of, a compilation of the most popular articles on The NeceMoon™.

The main objective of The NeceMoon™ is to share tips and insights on a sensible range of topics, in order to let others learn from my mistakes and successes, and hopefully to make things easier for the next person. The main objective of The Album is to promote an optimal access to that content. The Blog format does not always do justice to techno-strategic content. Originally, Blogs were designed in a journalistic spirit and are more suitable to chronological events and (more or less trivial) discussions around daily news. Even if the usefulness and the importance of an analysis persist in time, it becomes almost impossible to find and difficult to consult, as more articles keep stacking.

That is why the best articles have been cherry-picked, reviewed and arranged in a logical order that better matches the layout of a book. The Album is free of charge.

The NeceMoon™ can be accessed from NeceMoon.com (or necemonyai.com/blog)

The NeceMoon™ Album can be downloaded in various file formats, in full from TheAlbum.NeceMoon.com (or necemonyai.com/Blog/page/The-Album.aspx). The available formats are PDF, EPUB, MOBI/AZW3/KF8 (Amazon Kindle) and MP3. Furthermore, the various chapters and volumes can be downloaded independently/separately, according to your interests.

About the Author: who is Necemon Yai?



I am a Software Development Engineer extensively involved in Microsoft .NET technologies. Full time developer. Part time digital artist, strategist, essayist and entrepreneur. I majored in computer science from NIIT, Christ University and Swansea University (Master of Engineering, Computing).

At the time of publishing this, I have worked for a Europe leading E-Commerce company, a major UK financial group, the General Electric global corporation and a few tech start-ups that you probably never heard of.

Over the past decade or so, I have been running The NeceMoon blog, where I describe my experiences, my observations and my reflexions. I mostly talk about Technology and Strategy. Here I share my most popular articles.

About the Contributors: who is in your War Council?

I invited the best writers in my network to include some contributions in this book, especially some of their most relevant insights in terms of Technology and Strategy. These top authors are, Ahou The African Chick, Antoine Mian, Cyriac Gbogou, Darren Mart, Edith Brou, Holty Sow, Israel Yoroba, Jean Luc Houedanou, Jean-Patrick Ehouman, Karen Kakou, Monty Oum, Nanda Seye, Nnenna Nwakanma, Olivier Madiba, Vanessa Lecosson and Yehni Djidji.

Along with their respective writings, you can find links to their own web pages. In addition, most of these contributors introduce themselves and provide you with some tactics in our exclusive interviews that you will also find in this book.

About You, Dear Reader: who is this book for? What's in for you?

In The Album, there are 9 chapters organised in 2 volumes. Each chapter deals with a specific topic. You don't have to read everything. If you are interested (to one extend or another) in one or more of these topics, you would possibly appreciate the related chapter(s):

Volume 1: Moon Light (softcore)

- Chapter 1: Strategy and Tactics
- Chapter 2: Digital Marketing and Web Visualisation
- Chapter 3: Corporate Worlds and Emerging Markets
- Chapter 4: Quick Wins, Tricks and Tips
- Chapter 5: Transition - Extra Thoughts and Sharp Fantasy

Volume 2: Full Moon (hardcore)

- Chapter 6: Software Development and Engineering
- Chapter 7: C# .NET Programming
- Chapter 8: Epic Prototypes, Classic Projects, Historic Genre
- Chapter 9: Research and Case Studies

If you want to, you can download and read only the chapter(s) and volume(s) that you are interested in. Several file formats are available on TheAlbum.NeceMoon.com (or necemonyai.com/blog/page/The-Album.aspx)

All the Web links in this document are working, feel very welcome to click on them.



For All Practical Purposes

This document contains the Chapter 8 of The Album: “Epic Prototypes, Classic Projects, Historic Genre”. If you care, 8 other chapters are also available. Depending on your interests, you may download, (re-)read or share any of the various chapters and volumes independently/separately. The PDF, EPUB, MOBI/AZW3/KF8 (Amazon Kindle) and MP3 formats are available.

To get them, just click on any of the links you like below or go to TheAlbum.NeceMoon.com (or necemonyai.com/Blog/page/The-Album.aspx)

The NeceMoon Album (complete)

Volume 1: Moon Light (softcore)

[Chapter 1: Strategy and Tactics](#)

[Chapter 2: Digital Marketing and Web Visualisation](#)

[Chapter 3: Corporate Worlds and Emerging Markets](#)

[Chapter 4: Quick Wins, Tricks and Tips](#)

[Chapter 5: Transition - Extra Thoughts and Sharp Fantasy](#)

Volume 2: Full Moon (hardcore)

[Chapter 6: Software Development and Engineering](#)

[Chapter 7: C# .NET Programming](#)

[Chapter 8: Epic Prototypes, Classic Projects, Historic Genre](#)

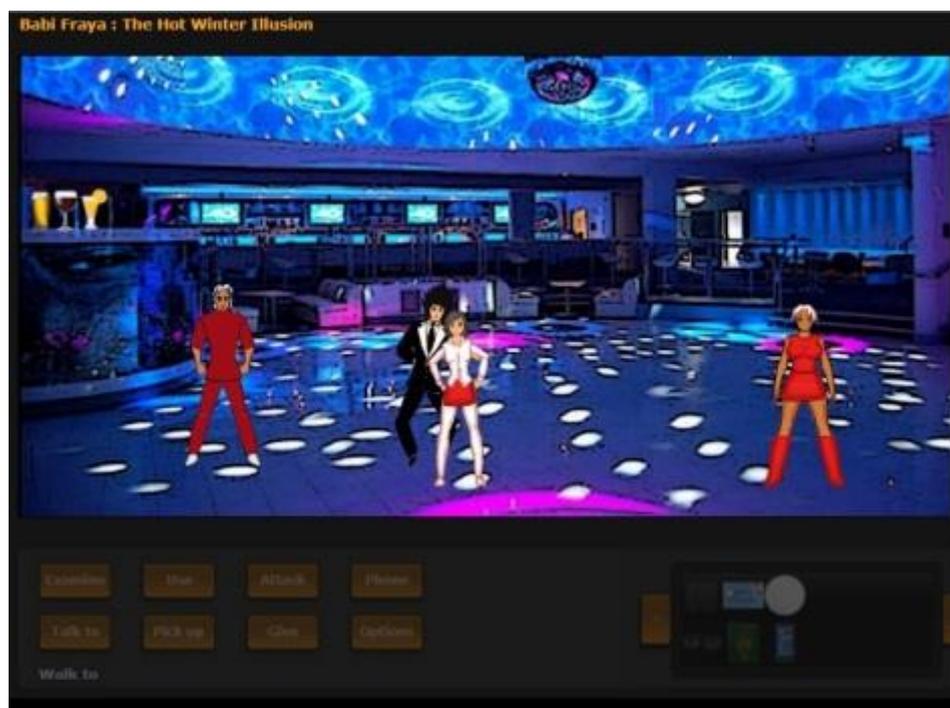
[Chapter 9: Research and Case Studies](#)

The Album is available in French as well at Album.NeceMoon.com (or necemonyai.com/Blog/page/L-Album.aspx)

Chapter 8

Epic Prototypes, Classic Projects, Historic Genre

necemon.com | babifraya.com | evasium.com | kpakpatoya.com



Inside Bavardica - Part 1 : Discussion of the subject areas

By [Necemon](#), 2010

A modern instant messenger is composed of various features which include public conversations, private conversations, user picture and signature. When it's time to innovate and make an instant messenger that is so special that it is both interesting and standing apart from whatever program already available on the market, it is not too obvious to figure out what extra features could be attached. It feels like almost everything had been already thought of and implemented in some other application.

However, [Bavardica](#) is my attempt to make a communication platform like no other. Bavardica, derived from Bavardage, which actually means talking in French, has got mostly two assets over regular instant messengers. Firstly, there is the permanent presence of artificial beings that interact with other users. Secondly, the application relies on a two-dimensional space which represents a small world in which the users are inhabitants represented by animated characters. This project is a good chance to set the scene of what could become an innovative virtual world.



Many areas of web entertainments are involved in the Bavardica Project. In its initial form, Bavardica was basically a chat application, but as it is evolving, Bavardica also fits in the following fields.

Artificial Intelligence

As taught to me by [Dr Phil Grant](#), the main objective of AI (Artificial Intelligence) is to try to make computers perform tasks that humans tend to be good at. The actual name “Artificial Intelligence” was coined by John McCarthy in the 60’s. There are a lot of topics which might nowadays be considered as part of the general subject of Artificial Intelligence. But for our current purpose, we shall only consider the themes of natural language understanding and representation of knowledge.

Now, what does it mean for a machine to be intelligent? Alan Turing thought about this problem in the 50’s and came up with the following test: A human would ask questions to both a computer and a human in some other rooms; if the interrogator is not able to figure out (from the answers) which is the machine then the computer is said to behave in an intelligent manner. A very early attempt to build a program which would pass the Turing test was the ELIZA program (by Joseph Weizenbaum). ELIZA was mimicking a conversation with a psychologist and was quite impressive at first glance. However, it was missing a common sense and an actual understanding of natural language (which are both pretty hard to acquire from a computer point of view). Nevertheless, many programs of this kind followed. They were based on tricks like:

- o The repetition of user’s statements verbatim (subject to pronominal adjustments)
- o Preceding the repeated statement with the introductory “Why do you need to tell me”
- o Asking “Why do you ask that”, which, in effect, change the topic or the level of the conversation.

Later on, Colby and AI wrote a program called PARRY based on ideas in ELIZA which had more complex properties like elements of emotion and more knowledge of grammar. It was intended to mimic a paranoid patient who believed he was being persecuted. In a real experiment, 33 psychiatrists were asked to rate the degree of paranoia of patients who were communicating through typed transcripts. Three of them were from human patients and two generated by PARRY. None stated that they thought there was anything unusual. Later the psychiatrists were told that some of the transcripts were machine generated, but they were, on the whole, unable to recognise which ones were from PARRY. Of course this whole situation is a bit odd anyway, as human paranoids behave in strange ways. The point here is that it may be difficult to differentiate a human talk from a machine talk (Phil Grant).

Moreover, throughout the time, AI systems are becoming more and more advanced, in fact, more intelligent. Currently, one of the strongest programs of this type is ALICE (Artificial Linguistic Internet Computer Entity) by Richard Wallace. According to Wikipedia, that’s a program inspired from ELIZA that engages in a conversation with a human by applying some heuristical pattern matching rules to the human’s input. The program uses an XML schema called AIML (Artificial Intelligence Markup Language). I use a similar model for implementing the AI feature in this project.

Virtual worlds:

A virtual world can be defined as a synthetic reality, an environment that is completely generated by a computer system and that is completely transportive in that the participant controls an avatar immersed in an artificial world, according to the book *Networking and Online Games* (Armitage, Claypool and Branch, 2006). It says that virtual worlds are intended for his users to inhabit and interact. Sometimes, the user can also manipulate elements in the said world and therefore experience a kind of telepresence. As the rate of people using virtual worlds is increasing by 15% every month and as there are nearly 600 million people worldwide registered in virtual world today (according to K Zero, a virtual word consultancy service), I felt that it would be a good plan to learn about this trend and give it a try. In its current form, Bavardica is a virtual world in the sense that each bavard (avatar) is aware of the other bavards and of the scene boundary.

Multimedia:

In a web browser, operations like timing, streaming, playing music forth and back, playing a list of music files etc. are not at all trivial using basic html and javascript. One big advantage of a browser plugin like Silverlight is that it gives much more ease about operations on media files. The book 'Accelerated Silverlight 3' had been really valuable by teaching me how the `System.Windows.Controls.MediaElement` control provides media playback capability. That control can handle both audio and video in a variety of formats like mp3, wmv, asx, etc. There is even a support for third-party codecs and for new media formats such as H.264, Advanced Audio Coding and mp4. Furthermore, the media element in Silverlight has a lot of properties, methods and events that allows a lot of control over the media. Some of the properties are `AutoPlay`, `CurrentState`, `CanSeek`, `IsMuted`, `Position`, `NaturalDuration`, `NaturalVideoHeight` and `Volume`. The methods of the `MediaElement` control include `Pause`, `Play`, `SetSource` and `Stop`. Some of the events are `CurrentStateChanged`, `MediaEnded`, `MediaOpened`, `MediaFailed`, the later allowing some more error handling in case the media cannot be found or in the case the format is incorrect.

This control allows us to bring a symphonic soul to this project. So another aspect of the Bavardica project is the audio streaming, that is now in its most basic form with an mp3 music file playing in the background during the conversation. There are also sound effects that serve as alerts informing that a new user just logged in or that someone just left a new message.

We will now proceed and [describe how the application has been build](#).

Inside Bavardica - Part 2 : Progress and Achievement of the Project

By [Necemon](#), 2010

As we discussed the [subjects areas](#) related to [Bavardica](#), we can now go through the steps in the actual achievement of the project.

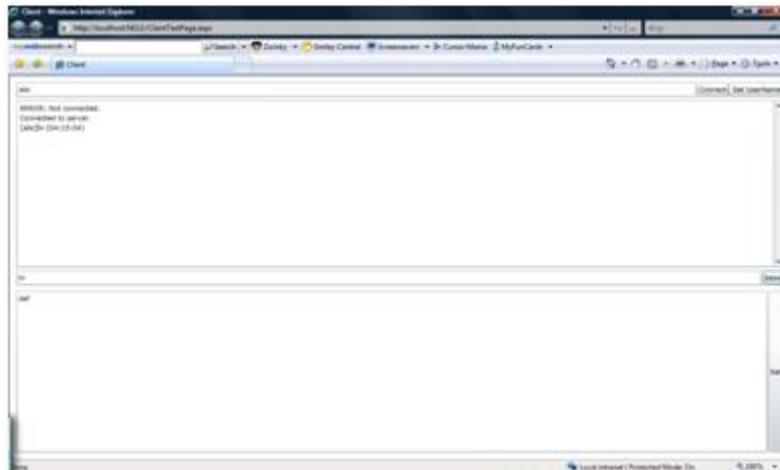
1. Sending a public message over the TCP network from a Silverlight client

The model selected for this project is the prototyping model. Starting first with a throwaway prototype, I could first send simple TCP packets using Silverlight and a console application as a server. It was a very basic chat that let a user, say A send a message to a user B so that no one else will receive the message apart from A and B. The server is a .Net console application and the client is a Silverlight application hosted in an ASP.NET page.



```

file:///E:/Projects/Meng/Bavardage/Bavardage/bin/Debug/Bavardage.EXE
Policy server started.
Bavardage server started.
Press Enter to exit.
Received policy request.
Server policy file.
Messenger client #1 connected.
abc joined
Sending to abc and def
  
```



2. First contacts with Silverlight animation

The second program I wrote was a simple Silverlight application with an animated 2d character walking around a park that was actually the background of the web page. This experiment had for sole objective the study of the motion of a character over a 2D scene.

The first key point of this prototype is the way the avatar is animated when a direction key is pressed:

The second key point is that the model used a sprite sheet or frame by frame animation. It consists of sliding horizontally a sequence of adjacent frames with a transparent background, one frame at a time when a directional key is hit.

This approach used `KeyAnimationUsingKeyFrames` but will be dropped later in the project as it involved no interpolation and that it would have required a lot of time drawing each character in every single position.



3. Making the server lies silently in the background (As a web service)

It was planned to integrate the two prototypes so that users could both chat and move around the 2D scene. It was also planned that the server application lies silently in a web service so that a console application need not be kept open throughout the operation. Both goals have been reached in the next prototype.

The fact is that I thought about building a more robust server application based on which I would build an evolutionary prototype that will gradually grow with the project. That server had two main requirements:

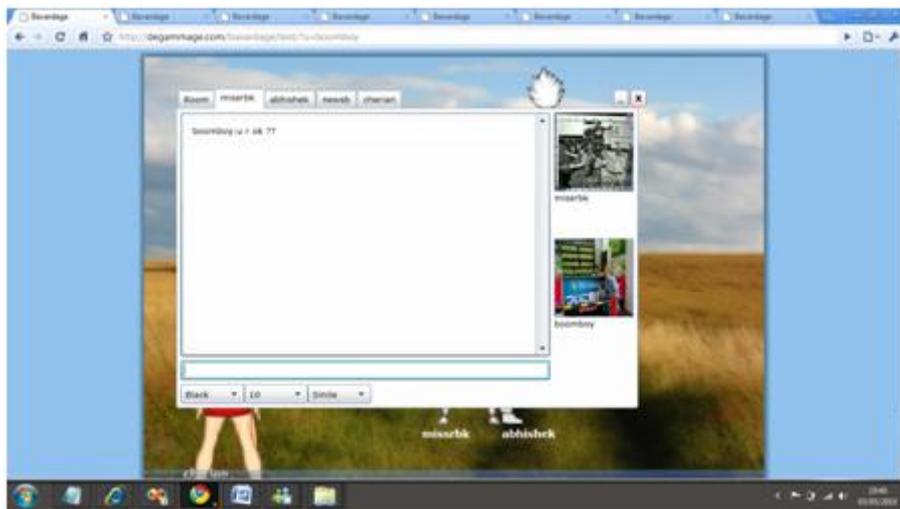
- It had to get rid of the console application. It should just be a silent process in the background. For this purpose, I just used web services.
- The server had to push data to the clients. That is sending data even when they are not requesting. In the classic HTTP model, first the client makes a request to the server and then only the server can respond. A solution to this issue was to use the WCF (Windows Communication Foundation) Duplex web services; and that's what I did. That's an asynchronous two ways communication model.

It's worth observing how the [WCF Duplex web services work](#)

4. Focus on One-to-One Chat

While the public chat happens in a common window, one-to-one chat involves an individual window for each conversation. The organisation of multiple conversations in Bavardica comes down to tabs, using the Silverlight tab.

The Room tab is the one used for public conversations. To start a private conversation, a user just needs to click on the name of the person he wants to talk to (There is a list of the available usernames right in the public room).



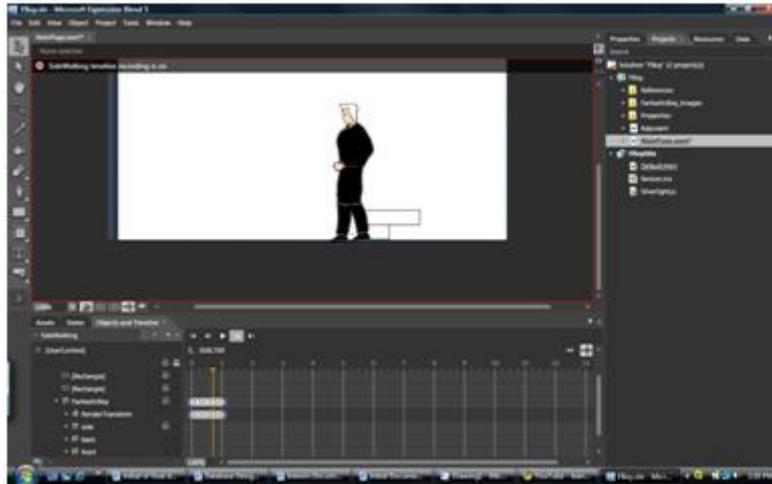
5. Drawing and animating characters

The next step was the design and animation of 2D avatars. There was already one avatar in the previous prototypes but it is not fun if everyone looks the same in a virtual world. Further, that particular avatar had been borrowed from some other Silverlight developer (Darren Mart). The previous model presented a lot of disadvantages. Besides the fact that a sprite sheet animation doesn't provide interpolation between frames, that model also involves a lot of drawing. Each individual position should be designed. But the biggest issue in our context is the nonexistence of customisation in that model: one of the first non-negotiable goals in this project was the ability to mix and match clothes and other outfits as well as their colours. This task is quite simple in a 3D world. It is easier to just swap meshes and textures but in a 2D world, it's a bit tricky to change a part of the outfit, unless we choose to draw a sheet for every conceivable combination of outfit pieces. The solution implemented in Bavardica was rather to create a set of baseline images that would collectively form a character. Therefore I ended up drawing my own characters.

So even if this is mainly a programming project, and not a graphic project, it's worth mentioning that quite a lot of time and effort goes in drawing and animating original characters. Customization is also an important feature of a virtual world. Users should be able to choose how they look and how they are dressed. For this project, I chose to draw four basic characters (2 males and 2 females) and at least two costumes and hair styles for each character. Due to the time constraint again, I could not draw and animate more than that. The operation involves fully drawing and painting the bodies then drawing clothing over. Three basic skin colours are available and the users can change the colours of their clothes and hair. Here is a preview of one of the character:



The characters are basically animated to walk, to sit and to stand. The problem with animations is that it had to be done every time from the three angles (front, back, side). There is also an animation to make them look alive even when they are standing idle. Those models used the `DoubleAnimationUsingKeyFrames` model by opposition to the previous one who used `KeyAnimationUsingKeyFrames`. Microsoft Expression Blend had been used to process the animations:



There are two main advantages to the animation system that had been used:

- Silverlight's interpolation makes things silky smooth
- The characters are customisable. They can just shift boots or hair style (or anything else) as the users want.

Unfortunately, handling characters in a virtual world is not only about design. It involves a lot of coding. Let's start with the creation of a character.

A few array lists are maintained, offering a variety of outfits to the user, as well as shapes and genders

According to the choice of the user, the relevant parts are exposed and the parts that are not relevant are hidden. However, the changes in colour follow another logic.

What happens here is that the source file for each part of the body is fetched from a different directory that changed according to the choice of the user. Later in the application, when the user can change colours about all the parts of their outfits, the same logic is applied.

This was about the character creation. Once the character is created, details about its outlook are stored in the database as expressed in the database design. From there, every time the user comes online, the server can send all its details to all the other clients. As soon as a client is aware of another user online, he has to represent the relevant character in his scene according to the data that had been submitted:

Setting the other characters details work as in the character creation. It's mainly about hiding and showing images.

Now that we have seen how the code involved in characters appearance works, we will now attack the code involved in animation and motion. It starts with a key pressed by a user that is handled by Silverlight. In our example, we will consider the right key:



6. The Cybavard

After the character design and animation, an emphasis has been given to the Cybavard, the artificial intelligence part of the project. The logic behind the Cybavard is inspired from the A.L.I.C.E bot introduced earlier in this presentation. What is interesting about A.L.I.C.E is that it was based on an XML dialect called AIML (Artificial Intelligence Markup Language). Due to the universal nature of XML, it can be used with a wide variety of language. The point is that, as this is a .Net project, AIML could also fit to the Cybavards. It was all about finding a C# AIML interpreter. For this project, I used a .net library file, AIMLBot.dll (by Nicholas Toll), released under the GNU General Public License. That program offers a lot of advantages:

Very small size: about 52 k

Very fast: can process 30 000 categories in under one second

Means of saving the bot's brain as a binary file

A simple and logical API

The inner working of this program is also fairly simple. It's based on pattern matching: for each sentence entered, it checks in its knowledge base which stimulus response would fit best:

So what I am doing is adding a reference to the binary file from my web service, I also upload the knowledge base on the server. The knowledge based is mainly a set of AIML files that I can edit as per my requirements. I could also add my own AIML files. From the cybavard class, what I had to do was creating a bot object and loading the AIML files.

In the following part, we will go through [the design of the characters](#).

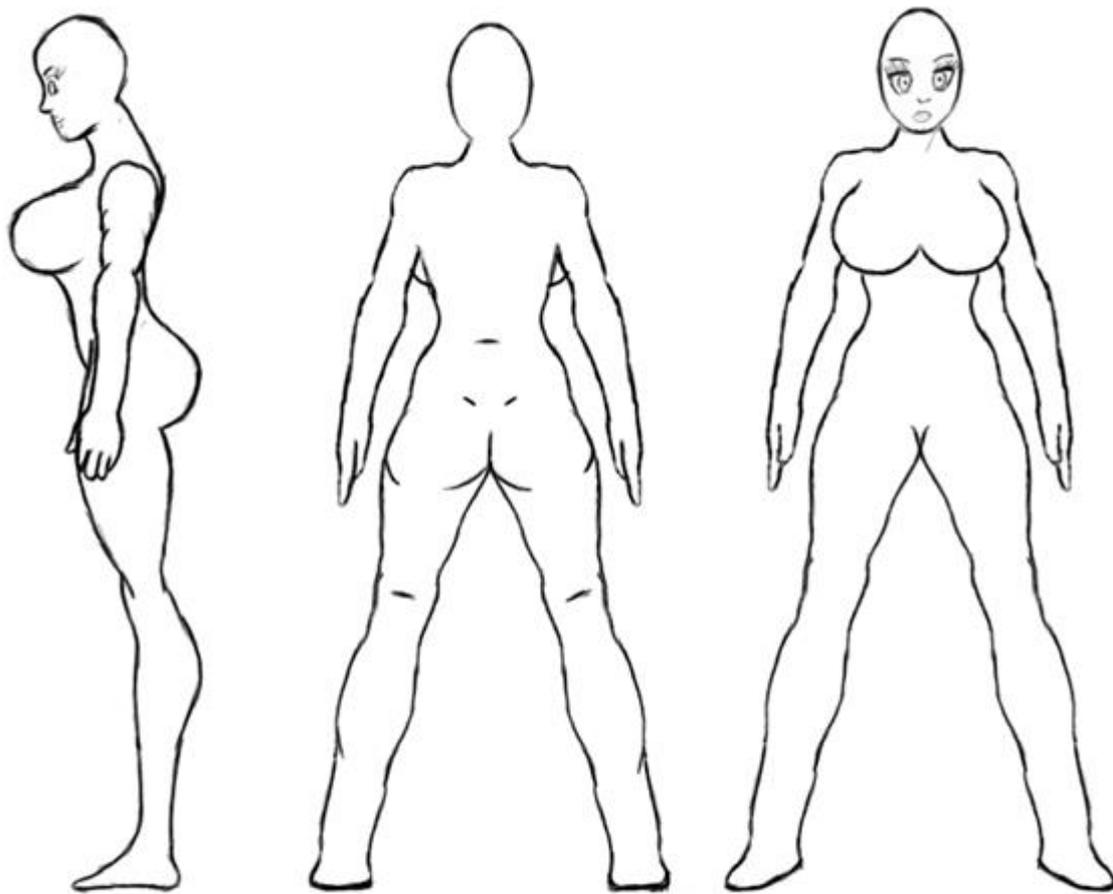
Inside Bavardica - Part 3 : Design of the 2D characters

By [Necemon](#), 2010

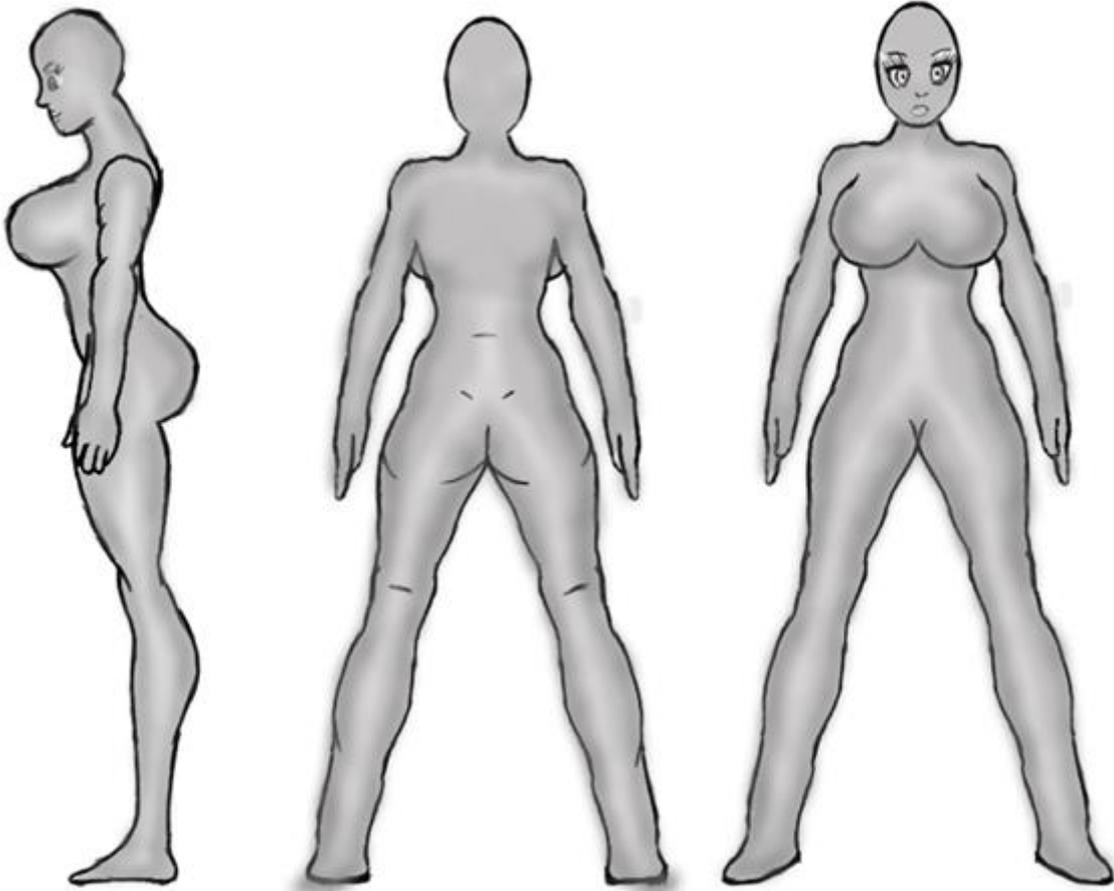
How I draw the 2D characters

Due to the nature of the drawing art, which is not really computer science related, I thought it was best to describe this process in another part right after [the progress description](#). This work had been a full part of the project and even though it does not directly concern programming (before we get into animation), it is also a computer work involving a pen tablet and some image processing software (Adobe Photoshop). The following screenshots describe the steps in the drawing of one character (out of four).

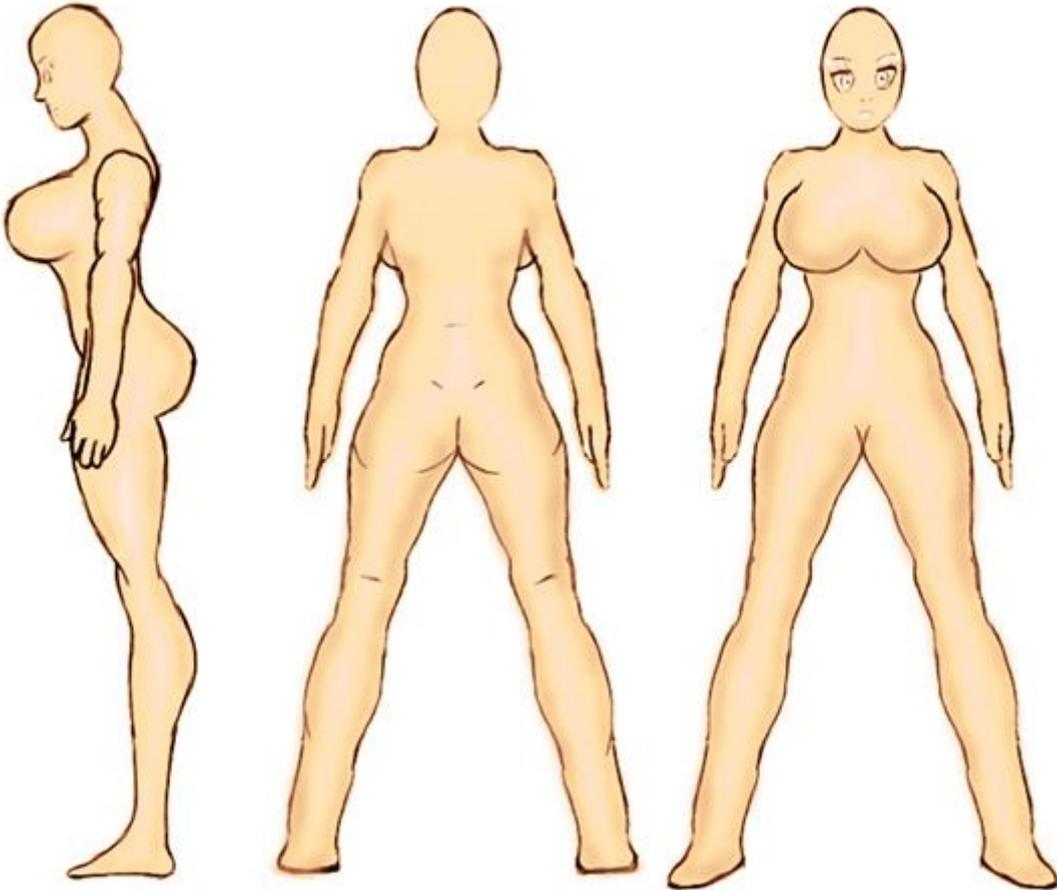
A. Sketching



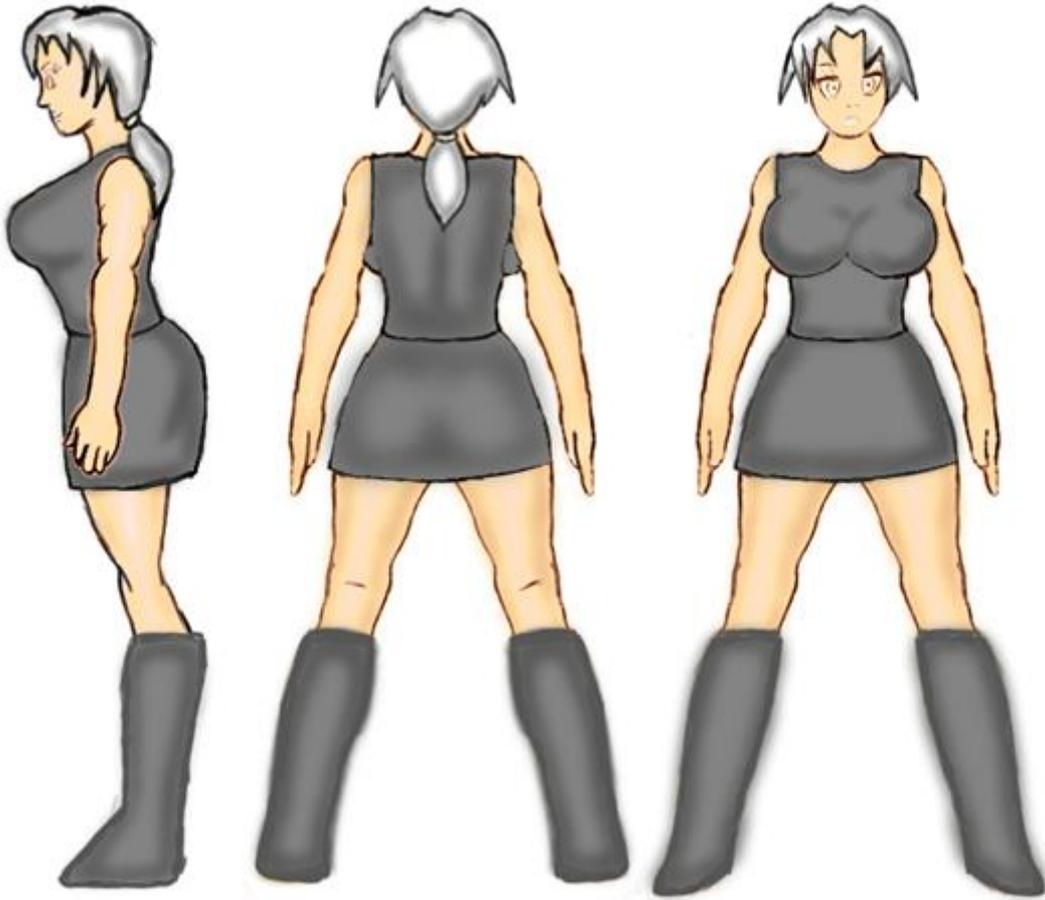
B. Filling



C. Coloring



D. Clothing



For details about animation techniques, you can check [those tutorials that Darren Mart wrote](#). To check out what's coming next with Bavardica, [move on to the next part](#).

Inside Bavardica - Part 4 : Scope for future improvements

By [Necemon](#), 2010

This is the last part of the [series](#) on [Bavardica](#). Apart from a user friendly and colourful user interface, the other features include the edition of the [character outlook](#) in real time (while on the scene). Here again, hair style, skin, clothes and shoes can be changed. Additionally, colours of all those garments can also be changed. The way it works is pretty similar to the way a character is created.

In the meanwhile, some other functions had been added:

Collision detection: This is to make sure that bavards don't get out of the scene boundary. Also, they don't walk inside each other when move vertically. This feature is based on simple "if conditions" that check whether the move doesn't not lead outside the scene or at a position occupied by another character.

Bubbling: That function is used to process text entered in bubble (graphic mode). Every time a user sends a message, it appears on the scene as a flying bubble that gradually disappears. An animation had been designed for this purpose. So both motion and opacity animations are applied to the bubble object. Each key pressed is processed individually as the bubble does not rely on a textbox. The main issue in this task was that the positions of special characters change according to the keyboard type. Finally the bubble handles pretty smoothly conversations from American and british keyboards.



More rooms

The current application contains a single scene where all the public conversations happen. A good extension could be to have many scenes and a map to navigate among the scene.

More music

The music background is always the same mp3 track playing all over again. It is intended in the next version to have a playlist of tracks as music background. Further, since there will be many rooms, each room will have a different theme and a different kind of music.

More cybavards

Due to the large amount of time and data required to simulate the talk of a single cybavard, it has not been possible to provide more than one of those cyber bavards. The process in their creation is pretty simple. It's mainly a matter of time. For instance, the cybavard in this demo has a knowledge base of nearly 4 megabytes (of text).

More animations for a better interaction

The only animations available at the moment are walking in all 4 directions and breathing while the character is idle. It is planned to add more animations like dancing, running, jumping. There will be some moves to express the character's mood. At a later stage, there could be animations that involve more than one character, like hug or hand shake. It is also planned to have more animations for characters to express their moods.

A working version is available at <http://bavardica.com>

10 things I learned from building Bavardica

By [Necemon](#), 2010

This is just a summary of the most important lessons I learned from building and publishing Bavardica. I sincerely hope you get some useful information (or reminder if you already knew all of this) from that. (PS: a working version of bavardica is available at <http://bavardica.com>)

1. **Nobody will use an application unless it provides some added value**, something interesting they don't already find in what they normally use. This is so true, even for free applications. They might try it but they would give it up quite soon if there is no actual incentive.

2. **The very fact that they are asked to register will discourage many people from trying a web application.** The most obvious causes might be that they don't want to waste time doing it, or they don't want to give away their details for privacy purpose. So at least a trial version should be offered to visitors without registration so that they can see what it's all about before they make any decision.

3. **The various commands and features are (only) obvious to the developer who built the application.** However, they may appear as strange and complicated to use for some users. The application should therefore be as simple as possible and there should be some hints for every command (right inside the application, not on an external file).

4. **Google is your friend** (well, I guess search engines in general). I spent a lot of time reading about Silverlight at the beginning of the project but whenever I had a doubt, I relied more on web searches than on books. I could find most of the answers I was looking for in technical forums. I could also find related work from senior developers (mostly [Anoop Madhusudanan](#), [Tomasz Janczuk](#) and [Darren Mart](#) in this particular occasion).

5. **Client-server architecture and silent web services.** This one is purely technical. I learned how to make an application lies silently in the background as a web service or a windows service; And I learned how to push data to the client over HTTP using a WCF Duplex Polling service.

6. **The more choice, the merrier.** Users enjoy having a lot of content to choose from. They like to express their personality and. to customize their character. Art and media requirement consume much more time than actual coding, but isn't it worth it? In the coming versions I would definitely be adding more rooms, more clothes and more animations. But...

7. **Not all at once.** It's important to move one step at the time not to get overwhelmed. It's all about priority management.

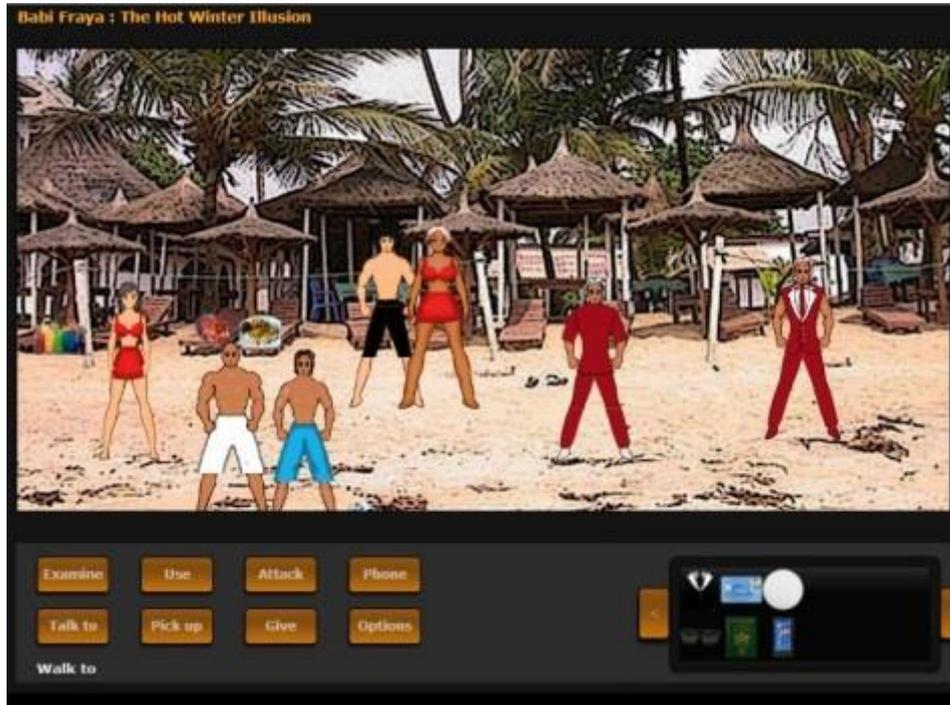
8. **Work results can be rewarding.** I am mostly talking here about the satisfaction and sense of accomplishment one's gets from work. It feels good to accomplish something, even if it's just a small thing for now. It motivates me to keep going.

9. **Feedback is priceless.** I could not learn all those lessons I am describing in these posts if I didn't have some guys and girls to show me my mistakes, to tell me what they didn't like about my work and how I could improve it. I would like to take this chance to thanks all those who tried Bavardica. Thank you for your feedback.

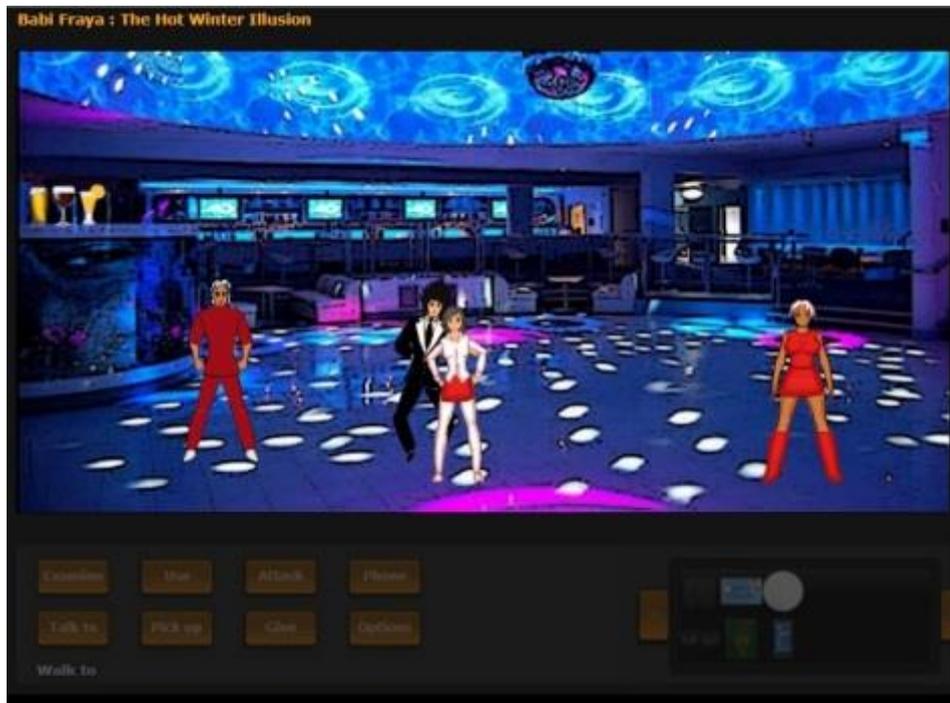
10. **There is a long way ahead** (to build a decent virtual world).

Babi Fraya: The Hot Winter Illusion

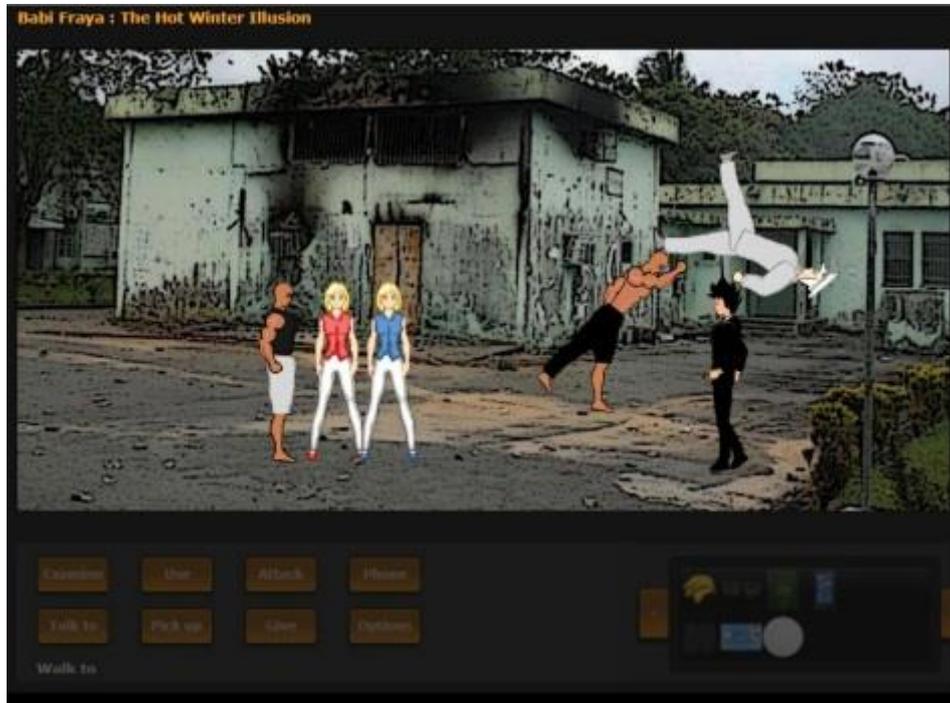
By [Necemon](#), 2013



I just published a beta version of my [first complete game](#). If you have a moment to try it, I hope you will enjoy it : <http://babifraya.com/>



Babi Fraya™ is a "[Point & Click](#)" [adventure game](#), therefore a game in which the player assumes the role of protagonist in an interactive story driven by exploration, dialogs and puzzle-solving. It's also a fiction which tells a story, even though the player can affect the story, as there are different screenplay branches.



Scenario

Winter 2010. Just another random New York student heading back to Babi where he is originally from. Great time ahead for Lewis ! Partying all night in Rue Princesse (Princess Street), enjoying the garba on the sunny beaches, hanging out with old buddies and new encounters... Until a sudden change of atmosphere. What could have possibly gone wrong? Nobody knows. It all went really fast... Within a few minutes everything went from harmony to drama: screaming, bombing, looting, violent clashes, food shortage, closure of borders. When Babi Joy becomes Babi Frayeur, what was meant to be a dream holiday increasing looks like a commando mission. What to do now? Fight or Fraya? Fight to Fraya? [You choose](#).



In short, I just have 3 things to say about it:

- This is a [point-and-click adventure game](#), built on [Microsoft Silverlight](#).
- The story offers an off-the-wall perspective of the [#CIV2010 Babi Battle](#), turning dream holidays into a commando mission.
- The game is quite easy and a few parts could potentially be improved but it works fairly well...

Babifraya.com. Any feedback is welcome,

N.

Relaunch : Kpakpatoya 2.0 as an Interactive Platform

By [Necemon](#)

#Kpakpatoya

1. Introduction

[Kpakpatoya™](#) is a repository for what's new, popular and relevant on the Internet with a focus on Africa. Users like you can submit content and decide, through voting, what's good (kiffs) and what's junk (zaps). Links that receive community approval raise towards the top, so the front page is constantly in motion and (hopefully) filled with fresh, interesting links. Kpakpatoya™ is part of the [Evasium Network](#).

"Kpakpatoya" is a term from the Ivorian slang which indicates the act of spreading rumors, passing on supposed stories relating to surrounding things, events and people. In our context, it would therefore be appropriate to understand "kpakpatoya" with regard to scoops, gossips and even compelling news.

Kpakpatoya is a catalog
for what's new, popular and
relevant on the Internet
with a focus on **Africa**.



kpakpatoya.com

pop in, they'll tell ya
#Kpakpatoya

2. Historical Context

April 2011. The Ivory Coast was then emerging from a major crisis, a crisis in which information and communication technologies have played an important role in providing practical assistance to related casualties. Beyond simple commenting, hashtags such as #CIV2010 and #CIVSocial were used to spread critical news and provide concrete solutions to these victims. Through donations, call centers and notably Twitter, which then comes up as the ideal tool for broadcasting urgent information in real time.

The Ivorian community on Twitter was growing and booming, with more and more registrations, tweets and contributions. But that community was still relatively small in terms of numbers. For such a small group on Twitter, how to get together, and exchange around common themes? Through the #Hashtags again, of course... On online social networks, the Hashtag is a word or set of words preceded by the symbol '#'. The hashtag is used to centralize messages around a specific term. It acts as a keyword so that users can follow, comment on a conversation or get together around a topic, for example the measures of refueling in times of crisis or even medical emergencies.

But in this post-war period, it was time to switch to less sinister tags. [#Kpakpatoya](#) was quite timely for this small community seeking to gather in an atmosphere that could be described as "cool". It was the tag that these Ivorian "influencers" and "pioneers" on Twitter used to do without local news, inside jokes, viral content, or other kinds of interesting information, including sports events, (technology) conferences, or day-to-day life.

Detecting a chance to improve some socio-technological conditions, I launched the [kpakpatoya.com](#) website in that same period. At the beginning, it was just [a flow of tweets around the #Kpakpatoya feed + a few extra features](#), including a summary of local and international news published in real time. Then I tried with varying degrees of success to offer some applications such as some discussion forums, the blog and the ability to tweet directly from the web site. I had also opened the Twitter account [@kpakpatoya](#) and arranged the mobile application implementation, which still present the news, as well as the Facebook page and the website.

Through this form of ecosystem, and a combination of technical and social skills, the hashtag #Kpakpatoya grew in popularity. By following the thread #kpakpatoya, it was easy to find CIV culture sympathizers, their news, as well as the most popular local themes, even for a newcomer.



3 years later, the Ivorian Web scene turned into a completely different form...

On the one hand, this virtual community has expanded a lot, which is most welcomed, since there is a lot more content, web projects, web entrepreneurs, bloggers, users, etc. Except that a simple hashtag is not enough anymore for an effective exploration of the content. It would not be practical to communicate around a single hashtag. Of course, there is always the option to "break down", to use the alternative hashtags that have emerged in the meantime (although #Kpakpatoya remains one of the most popular tags). That's a good solution, provided one can constantly be aware of new users and tags to follow, so as not to get lost in all this volume of content, and not to miss the most relevant and recent information..

On the other hand, there has been very little evolution when it comes to the #Kpakpatoya project, whether it be in terms of technology, computer graphics, communication, infrastructure, etc. The Facebook page has less than 2000 fans, the Twitter page has around 8000 followers, the mobile application is obsolete, and until recently, the components of the website that had a lot of success in the first few months have been almost neglected (I will spare you the numbers related to the decrease in web traffic). There are almost no contributors to the progress of the project, there are just the users. This could have been a decent community project, but instead it was reduced to a latent state.

It is true that the community was formed in a rather organic way, there was no concrete strategy, everything happened gradually and naturally, which perhaps explains the cool-down after the buzz effect had passed, each party being concerned with their own projects. I must admit that the applications I wrote as part of this project were essentially in maintenance mode and that I had moved on to other things myself. Anyway, no one stepped up.

A wise man once said: If there is anything potentially interesting and you do not understand why nobody does it, it's because you haven't done it either. That's DIY: Do It Yourself.

I therefore take over the lead on this project to try and produce something useful. In recent weeks, my team and myself have been doing some renovations, trying to produce a more interactive platform (with some added value, if at all possible) in order to facilitate meetings, exchanges and content management within the community.



3. Description

Kpakpatoya™ is powered by [Evasium®](#), a small team of technology enthusiasts. We basically make Internet software. We focus on building edutainment-oriented web/client applications.

On Kpakpatoya™, you can discuss pretty much anything, but the most welcomed topics are : Buzz, Clash, Economics & Politics, Culture, Insolite, Local News, Africa, Sport, Diaspora.

Beyond the option to read news, it is now a matter of adding a touch of interactivity, where everyone can share links, discuss, comment and vote on the existing notes. A Kiff is a positive vote (like a "like", a "+1") for a note. A Zap is the opposite, it is a negative vote for a note. Any Kpakpatoya user may kiff or zap a note, and since the score of the note depends on the difference between his kiffs and his zaps, each vote helps to establish the position of the note in the Kpakpatoya ranking. The notes could be listed in order of popularity, to allow the next visitors to spot the best content.

Nothing too complex, it's just a sharing system as found everywhere on the Web. For example, tech entrepreneurs have [Hacker News](#), Americans came up with [Reddit](#), French cybernauts have [tapemoui.com](#). The concept has existed for a while, it's just that there was no Ivorian/African version.

Kpakpatoya™ is available in 2 languages.

In English:

<https://kpakpatoya.com/?lang=en>
<https://www.facebook.com/KpakpatoyaNews>
<https://twitter.com/kpakpatoyanews>

In French:

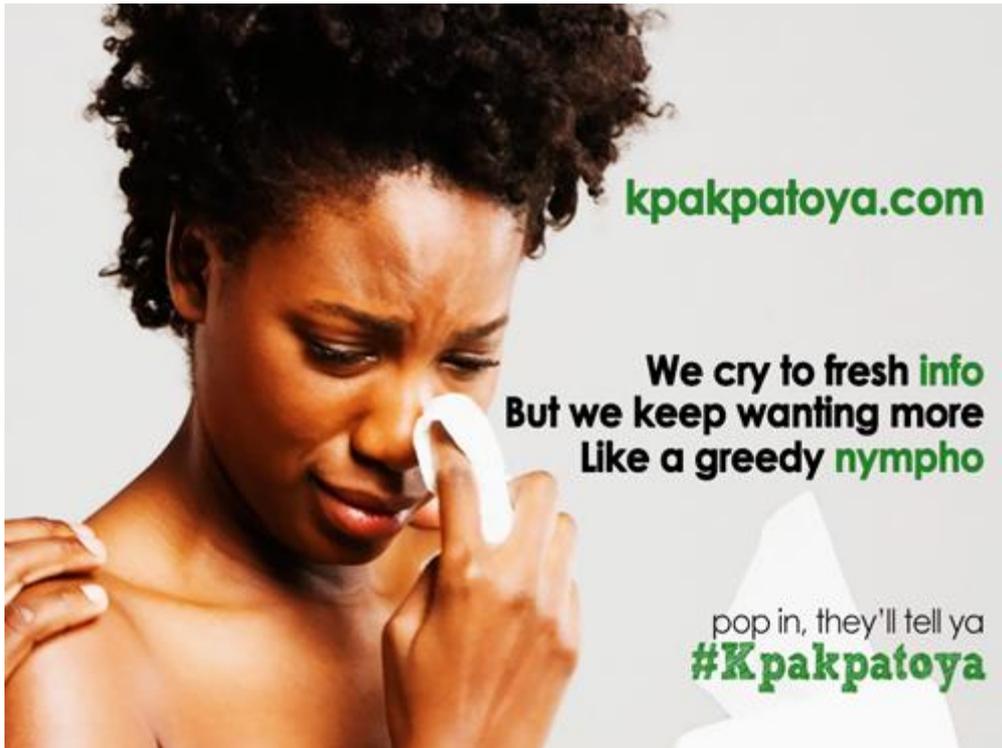
<https://kpakpatoya.com/?lang=fr>
<https://www.facebook.com/Kpakpatoya>
<https://twitter.com/kpakpatoya>

Kpakpatoya™ may (or may not) become a reference point in terms of news and "affaires" for Ivorian web users. But that's not the main goal here. The current goal is to provide a useful tool for expanding the sharing options for local content, and to implement the long overdue Kpakpatoya renovation.

On a more personal note, I think we should rather see this project as an experience that could potentially help to optimize the visibility of online African content.

For having managed the website, the Facebook account and the Twitter account of Kpakpatoya™ over the last 3 years, it seems to me that many followers expect a better interaction system and they strive for a better visibility for their content, namely their projects, links published, or simple opinions and comments. Therefore, it is possible that this version of the site meets their expectations to some extent.

There are still many things we could add or improve, we can do it as we go along, while taking users feedback into consideration, but the prototype of this new version is relatively stable, and what is needed now is trying it. That is to say, create your account, test the different forms of interaction, provide your feedback and share the link with your contacts so that they can do the same.



Flux Project : Social News Web Applications

By [Necemon](#)



This is just a quick update on my [ongoing side projects](#).

I have been working on a series of [social news web applications](#) where users can submit content and decide, through commenting and voting, what's good (kiffs) and what's junk (zaps). Links that receive community approval raise towards the top, so the first pages are constantly in motion and (hopefully) filled with fresh, interesting links.

Nothing impressive, but it gives me a good basis for making a robust web platform, as in, the common features:

- Security (cryptography, membership, login, registration, password reminder, etc.)
- Database layer / caching (or how to manage an ever growing volume of "big" data)
- Back end services (email senders, pings, http requests, logging)
- A stable [ASP.NET MVC](#) application
- User actions/management : voting, commenting, publishing, moderation
- Assets management (images, javascript, css, minification/bundling)
- Bonus: I am learning a lot when it comes to deployment (IIS settings, virtual directories, ports, application pools, etc.)

In short, the kind of things that could be useful in a modern [web application](#). Actually, this is my most advanced web application so far.

On [Extreme Programming](#) mode

I am still interested in building [virtual worlds](#) and [online games](#), but I thought I would release something (anything) at regular intervals. That way, I can improve as I go, learning/shipping at every step, instead of working on one thing for many years and come up with something that's not really practical/useful. So, that's the idea : getting stuff out there early, as soon as I've got something that covers the basics, and then build it up. If my experience with web projects has taught me anything, it's that people who use apps will suggest or ask for things that the developers wouldn't have thought of doing for themselves, and some of them turn out to be really good ideas even if we don't always quite see the potential at the time. Also it helps getting a better idea of priorities.

Anyway, that basic version might already be quite useful in some contexts, as news portal, one of the flavours being [kpakpatoya.com](#). you can think of it as a small bilingual Reddit for African news. It's starting to have a decent audience (with [8000+ followers on Twitter](#) and [2000+ fans on Facebook](#)) and some of the followers may welcome an opportunity to browse more content and to interact around it (share, comment, vote and add their own content).

So that's what I have been up to recently, each site having its own topics and they are all available online (in English and French) :

[flux.evasium.com](#) (Education, Science and Virtual Worlds)

[notiflux.com](#) (an exploration of online content on Personal Development and Winning at life)

kpakpatoya.com (a repository for what's new, popular and relevant on the Internet with a focus on Africa)

degamage.com (essentially geeky entertainment)

appuyage.com (an exploration of psychology, sexology and interpersonal relationships at the emotional level)

So [I did a game](#) the year before, shipping these web portals this year and the coming months sounds promising in terms of technical improvements and new features. Let's see.

[Stay up to date with your favorite topic on social platforms](#)

By [Necemon](#)



Having written [Notiflux™](#) and [AUDE](#) (Automatic Updates Distribution Engine) some time last year, I thought I would play around with the [Facebook](#) and [Twitter](#) APIs and keep posting links to some fresh news and articles in real time.

So depending on the kind of things you like, you might want to join or follow some of those pages:

[Degamage™ UK](#) : PC & Web gaming news.

->[Facebook](#) ->[Twitter](#)

[Evasium Park](#) : A [2D Virtual World](#). We are currently working on the version 0.2. In the meanwhile, here are regular news about virtual worlds.

->[Facebook](#) ->[Twitter](#)

[Notiflux™](#) : Your flux of notes and notifications. All about winning : strategy, tactics, philosophy, marketing, business, etc.

->[Facebook](#) ->[Twitter](#)

[Evasium®](#) : Posting news about fun learning, education through technology.

->[Facebook](#) ->[Twitter](#)

[Evasium Developers](#) : The nerdy one. Latest and best technology articles about programming in general and Microsoft C# .NET in particular.

->[Facebook](#) ->[Twitter](#)

Take your pick!

N.

Furtivue (Or How To Send Furtive Messages Like A Ninja)

By [Necemon](#), 2012



Furtivue : a furtive view over your messages. (www.furtivue.com)

Furtivue is a web service that allows you to send temporary (self-destructing) emails, as in, you can control how long the message is going to stay with the recipient.

When they open your Furtivue message, it appears for a specified number of seconds, then self-destructs. The duration will depends upon the length of the message (Or you can choose how long you want the message to be displayed).

How it works:

User A writes a message to user B, a link is sent to B. When B clicks on the link, the link is de-activated (therefore can't be used thereafter), the message is shown from a Silverlight canvas and disappear after the predefined time is over.

Purpose:

This could be used for sending confidential/secret information (such as passwords), or any other data you don't want the receiver to keep permanently in their e-mail box : for some reason, you may want to express yourself without leaving a permanent record.

Now what?

You can [try it here](#) or [download/explore the source code from here](#).

Final Reminder

Before we conclude, I just thought I would remind you that this document contains the Chapter 8 of the Album, and that 8 other chapters are also available. You may download, (re-)read or share any of the various chapters and volumes independently/separately, depending on your interests. The PDF, EPUB, MOBI/AZW3/KF8 (Amazon Kindle) and MP3 formats are available.

To get them, just click on any of the links you like below or go to TheAlbum.NeceMoon.com (or necemonyai.com/Blog/page/The-Album.aspx)

The NeceMoon Album (complete)

Volume 1: Moon Light (softcore)

[Chapter 1: Strategy and Tactics](#)

[Chapter 2: Digital Marketing and Web Visualisation](#)

[Chapter 3: Corporate Worlds and Emerging Markets](#)

[Chapter 4: Quick Wins, Tricks and Tips](#)

[Chapter 5: Transition - Extra Thoughts and Sharp Fantasy](#)

Volume 2: Full Moon (hardcore)

[Chapter 6: Software Development and Engineering](#)

[Chapter 7: C# .NET Programming](#)

[Chapter 8: Epic Prototypes, Classic Projects, Historic Genre](#)

[Chapter 9: Research and Case Studies](#)

The Album is available in French as well at Album.NeceMoon.com (or necemonyai.com/Blog/page/L-Album.aspx)

Conclusion

This is Good Bye. However, we can stay in touch. Feel very welcome to add me on [LinkedIn](#), [Twitter](#) or [Facebook](#). If you also have any tools and tactics that help you achieve things efficiently, I would like you to tell me about that. My email address is necemon@gmail.com. You are more than welcome to write to me and report any possible mistake in this document, or to suggest any improvement, or to tell me about your favourite parts. However, if you don't like The Album, don't bother writing to me.

Anyways, I wish you all the best for your ongoing and next projects. Thank you for reading. And many thanks to [Wikipedia](#), [MSDN](#), [IconFinder](#) and [FreeDigitalPhotos](#) for the clarifications and the graphic resources. Special thanks to the Evasium Team. Thanks to all those who contributed, thank you Ahou The African Chick, Antoine Mian, Cyriac Gbogou, Darren Mart, Edith Brou, Holty Sow, Israel Yoroba, Jean Luc Houedanou, Jean-Patrick Ehouman, Karen Kakou, Nanda Seye, Nnenna Nwakanma, Olivier Madiba, Vanessa Lecosson and Yehni Djidji.

Thank you Monty Oum, rest in peace.

[Share The Album with Your Mates](#)

Just click on any relevant icon below. It only takes a couple of seconds and it's free for everyone. Alternatively, you can share this link on any social media website or email it to your contacts that would benefit from reading it: <http://TheAlbum.NeceMoon.com>

