

Análisis e información adicional

Un diccionario; en XML, que permita importar nuevos ficheros a través de la conexión WiFi o del cable USB.

Katharina Maria Kaczmarczyk

katkac@ei.upv.es

Ingeniería Técnica Informática de Gestión (ITIG)

Director:

Manuel Agustí Melchor

magusti@disca.upv.es

Grup de Visió per Computaor (VxC)

Departament d'Informàtica de Sistemes i Computadors (DISCA)

Escola Tècnica Superior d'Enginyeria Informàtica

Código P.F.C: Disca-144

Valencia, 2011



Índice

Índice	i
Lista de figuras	ii
Lista de tablas	iii
Lista de abreviaturas	iv
1. Análisis y Diseño	5
1.1 Diferencias / cambios de la versión alfa	5
1.2 Diseño general	6
1.3 Logotipos	16
1.4 Mascota	19
2. Información adicional	25
Declaración jurada	31

Lista de figuras

Figure 1: Varias capturas de pantalla de las AA de NDS del perfil.	6
Figura 2: Varias capturas de pantalla de las AA de NDS del menú principal.	7
Figura 3: Varias capturas de pantalla de las AA de NDS de los diccionarios.	8
Figura 4: Varias capturas de pantalla de los diccionarios oficiales para el NDS.	8
Figura 5: Varias capturas de pantalla de los juegos de NDS de pantalla principal.	9
Figura 6: Ejemplo de la zona "activa" en el NDS.	9
Figura 7: Ejemplo de un diccionario con 2 zonas en una AA.	10
Figura 8: Ejemplo de la estructura del diccionario online "wordreference.com".	10
Figura 9: Ejemplo de un diccionario electrónico.	11
Figura 10: Comparación de las pantallas de PSP y NDS.	11
Figura 11: Ejemplo de cambio de resolución de pantalla de NDS con un videojuego.	12
Figura 12: Ejemplo de cambio de resolución de pantalla de NDS con una AA.	12
Figura 13: Captura de pantalla del aviso en el videojuego "the 3rd Birthday".	13
Figura 14: Captura de pantalla de menú principal del juego "the 3rd Birthday".	13
Figura 15: Captura de pantalla del aviso en el videojuego "Patapon".	13
Figura 16: Captura de pantalla de menú principal de "Patapon".	14
Figura 17: Captura de pantalla de menú principal de "the 3rd Birthday".	14
Figura 18: Ejemplo de la zona "activa" en "the 3rd Birthday".	15
Figura 19: Ejemplo de la zona "pasiva" en "the 3rd Birthday".	15
Figura 20: Primero logotipo.	17
Figura 21: Boceto de logotipo nuevo.	17
Figura 22: Logotipo del diccionario.	18
Figura 23: Ejemplo de los mascotas en las AA de NDS.	20
Figura 24: varias fotos de "white-faced-owl" de la búsqueda de google.	21
Figura 25: Foto de búho: "white-faced-owl". Referencia Youtube: http://www.youtube.com/watch?v=fRSCI1HWBC4	21
Figura 26: Model Sheet de búhos de Disney.	21
Figura 27: dibujos principales del búho.	22
Figura 28: Cambio de pluma "ceja" a pluma "oreja".	22
Figura 29: Esqueleto del "Profesor Tobí".	23
Figura 30: Emociones del "Profesor Tobí".	23
Figura 31: De boceto a color.	24
Figura 32: Animación del búho.	24

Lista de tablas

Tabla 1: Comparación de las especificaciones técnicas de las videoconsolas en el mercado. **Error! Bookmark not defined.**

Tabla 2: Ejemplo de lógica de visualización de un menú con 3 ítems.**Error! Bookmark not defined.**

Tabla 3: Comparación de los logotipos. **Error! Bookmark not defined.**

Tabla 4: Comparación de los logotipos de videojuegos. 17

Tabla 5: Resultado final de propio logotipo..... 18

Lista de abreviaturas

AA	Aplicaciones de aprendizaje
AM	Aplicación Multimedia
GUI	Graphical user interface
NDS	Nintendo DS
N3DS	Nintendo 3DS
OSK	On-Screen-Keyboard
PSP	Sony PlayStation Potable
SDK	Software development kit
SDL	Simple DirectMedia Layer
UPV	Universidad Politécnica de Valencia
VS	Visual Studio
WiFi	Wireless Fidelity
XML	eXtensible Markup Language

1. Análisis y Diseño

Como en la versión alfa, el diseño forma una parte importante en el desarrollo de la aplicación. Es imposible ignorar la importancia de un buen diseño, dado que todas las aplicaciones lanzadas en el mercado tienen sus “propias” características, sin embargo, la mayoría de los videojuegos para los dispositivos portables o videoconsolas portátiles poseen las mismas características básicas. Aparte de esto, el diseño debe ayudar al usuario a usar la aplicación sin dificultad. En particular para las videoconsolas, si una interfaz es demasiado complicada el jugador pierde el interés en el juego. Lo mismo pasa con una AA, adicionalmente si la AA es aburrida. Quiere decir, que no hay ninguna interacción entre usuario y aplicación. En concreto, la mascota interviene con el usuario, da consejos, habla sobre los resultados de los exámenes de la AA, etc.

En los siguientes sub-apartados de este capítulo, el objetivo consiste en aprender las características básicas del diseño de interfaces gráficas de los videojuegos actuales y AA actuales. Asimismo, aprender a diseñar una GUI con los resultados de análisis.

Fueron revisados los siguientes diccionarios y aplicaciones de aprendizaje para la NDS:

- My japanese Coach
- Langenscheidt Vokabeltrainer
- HMH Vokabeltrainer Spanisch
- HMH Vokabeltrainer Französisch
- Berlitz – English Coah
- Addy – Do you speak English
- Berlitz Basic Dictionary English
- Mein Wortschatz-Coach English
- English Buddy
- Practise English
- English Trainer

1.1 Diferencias / cambios de la versión alfa

Debido al análisis de los diccionarios para la NDS el traductor no era necesario, ya que es demasiado difícil traducir un idioma a otro idioma perfectamente, porque cada

idioma tiene su propia gramática y sus propias características. Además no es recomendable traducir un texto palabra por palabra. Por estos motivos, no existe un traductor en la versión actual. Otro cambio de la versión alfa a la versión actual consiste en el diseño de menús y teclados. Como comentado antes, el diseño de GUI importa mucho en la especificación de la aplicación y la valoración. Ambos teclados fueron cambiados para facilitar el uso y para hacer el diccionario más inteligente.

1.2 Diseño general

En este apartado la meta fue encontrar las características básicas de los videojuegos de PSP y de los videojuegos / diccionarios / AA de NDS. Especialmente las AA y los diccionarios fueron los más analizados, ya que los videojuegos en general no tienen que ver mucho con el desarrollo de un diccionario para la PSP.

Aplicaciones de aprendizaje / diccionario para el NDS

Todas las AA poseen de un personaje que ayuda al usuario con los ejercicios. Los personajes puede ser en 3d como en English Buddy o en 2d como en los ejemplos de lo demás. Las animaciones de personajes son muy simples, ya que los personajes solo mueven los ojos o la boca; lo que conlleva a que los juegos tengan un tamaño promedio de 100Mb, ideal para el desarrollo de nuevos juegos¹ para la PSP. Estas aplicaciones siempre requieren crear un perfil de usuario para los análisis de aprendizaje, como se puede ver en la **Error! Reference source not found..**



Figure 1: Varias capturas de pantalla de las AA de NDS del perfil.

Los menús principales de las AA siempre contienen los siguientes ítems, después de seleccionar su perfil:

¹ PSP Miniss son juegos de tamaño máximo de 100mb. Se los descarga por la página web de Sony networks.

- Ejercicios / exámenes de tipo Minis-juegos (o tipo de test)
- Estadísticas del estudio
- Diccionario, lista de los vocabularios aprendidos

En algunas de las aplicaciones, los ítems fueron divididos en sub-ítems. Generalmente son siempre tres tipos de ítems en la aplicación (estudio, análisis, historia / diccionario). En la **Figura 2** se puede ver los ejemplos de menús principales.



Figura 2: Varias capturas de pantalla de las AA de NDS del menú principal.

Los diccionarios de las AA son muy simples. Traducen una palabra o un conjunto de palabras² a una palabra de idioma destino. Algunos de ellos tenían un diccionario adicional de un conjunto de palabras en un campo especificado p.ej. animales: tigre, consejo, perro, gato, etc. o también una lista de los verbos irregulares. La **Figura 3** muestra unos ejemplos de los diccionarios de AA de NDS.

² Conjunto de palabras: por ejemplo la palabra inglesa "please" en español significa "por favor".



Figura 3: Varias capturas de pantalla de las AA de NDS de los diccionarios.

Los diccionarios oficiales para la NDS ofrecen muchas más traducciones de una palabra o un conjunto de palabras. Existen varios diccionarios para los idiomas asiáticos que ofrecen más al usuario que los diccionarios generales, es importante saber y aprender los caracteres asiáticos para poder entender textos. Para ver la escritura general de un diccionario oficial en la NDS, se muestra en la **Figura 4**.



Figura 4: Varias capturas de pantalla de los diccionarios oficiales para el NDS.

Videojuegos para la NDS

La mayoría de los videojuegos requieren que el usuario pulse la tecla "Start" para poder entrar en el menú principal. Hoy en día sobra este tipo de consulta totalmente, porque la tecnología ha ido avanzando y además es innecesario tener un "menú" con solamente un "ítem", como se puede ver en la **Figura 5**:



Figura 5: Varias capturas de pantalla de los juegos de NDS de pantalla principal.

Asimismo, la división de pantalla es muy extensible, ya que se suele trabajar siempre con la pantalla abajo con el "stylus". La de arriba sirve como mapa de ciudad / mundo o en algunas ocasiones como libro ilustrado para contar la historia del juego. Esa idea de división de la pantalla es muy útil para AA, porque se trabaja siempre en una zona fija y por lo demás el usuario tiene la posibilidad de leer el texto de apoyo en cualquier momento. Un ejemplo concreto se puede ver en la **Figura 6**.



Figura 6: Ejemplo de la zona "activa" en el NDS.

Especialmente para los diccionarios electrónicos es recomendable dividir la pantalla en 2 partes. Una de ellas sirve para la búsqueda mientras la otra para mostrar la palabra y sus traducciones y los ejemplos puestos. Si no, el usuario tiene que moverse entre las ventanas si ha ocurrido un fallo al escribir, como se muestra en la **Figura 7**.

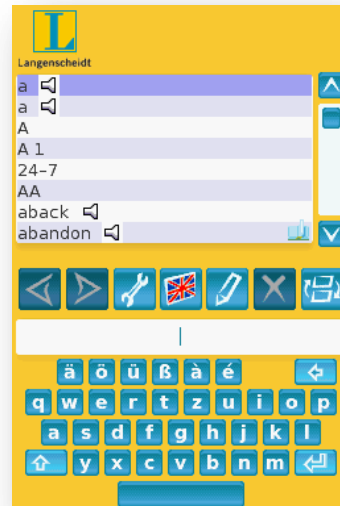


Figura 7: Ejemplo de un diccionario con 2 zonas en una AA.

También las páginas web de diccionarios tienen más o menos la misma construcción como las AA de NDS. Como se puede ver en "wordReference.com", **Figura 8**.



Figura 8: Ejemplo de la estructura del diccionario online "wordreference.com".

En cambio los diccionarios electrónicos de PONS no disponen de buscador integrado en la pantalla, lo que conlleva a otra problemática. El usuario siempre tiene que volver a la pantalla principal de la aplicación (**Figura 9**).

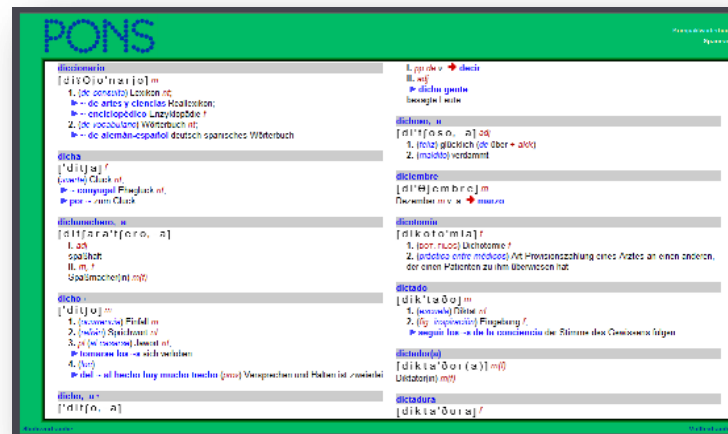


Figura 9: Ejemplo de un diccionario electrónico.

En comparación con la PSP, la pantalla de PSP siempre está “activa” en todos los momentos, esto es mejor para los videojuegos complejos en 3d, sin embargo para el desarrollo de un diccionario hay que “dividir” la pantalla en 2 partes. Una parte de ellas es la pantalla activa (para la búsqueda), la otra es la pantalla pasiva (mostrar la información) como en la NDS. Por ejemplo, si se ajusta la pantalla de abajo de la NDS colocándola a la derecha de la pantalla arriba, la resolución corresponde más o menos a la resolución de pantalla de PSP. En concreto la resolución nueva sería 512 por 192 pixeles (2*256 por 192). En cuanto a la resolución de la PSP, hay una diferencia de 32 por 80 pixeles. El cambio y las diferencias de la resolución se pueden ver en la **Figura 10**.

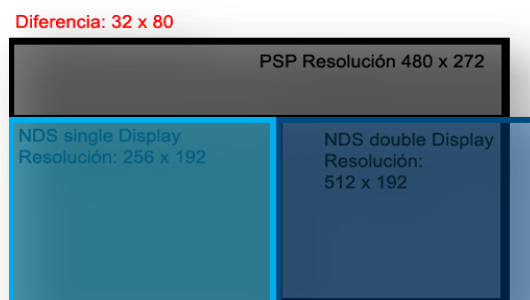


Figura 10: Comparación de las pantallas de PSP y NDS.

Para entender mejor el cambio de resolución de pantalla de NDS, se ha creado las siguientes imágenes con ayuda de un emulador, se muestra en la **Figura 11**.



Figura 11: Ejemplo de cambio de resolución de pantalla de NDS con un videojuego.

El mismo ejemplo en el caso de una AA se puede ver en la **Figura 12**.



Figura 12: Ejemplo de cambio de resolución de pantalla de NDS con una AA.

Videojuegos para la PSP

Algunos de los juegos requieren que el usuario tenga que pulsar la tecla Start” para poder entrar en el menú principal. Como se ha comentado en el apartado anterior. Además hay juegos que avisan al usuario de que el juego necesita un poco de memoria. La **Figura 13**, **Figura 14** y la **Figura 15** lo muestran.

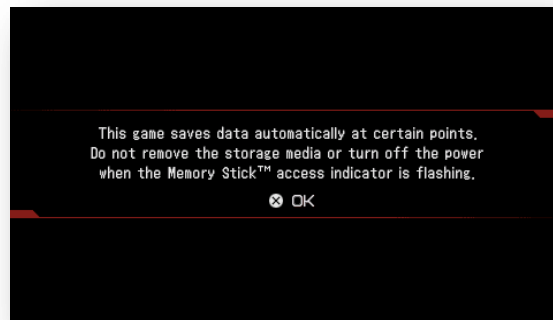


Figura 13: Captura de pantalla del aviso en el videojuego "the 3rd Birthday".



Figura 14: Captura de pantalla de menú principal del juego "the 3rd Birthday".

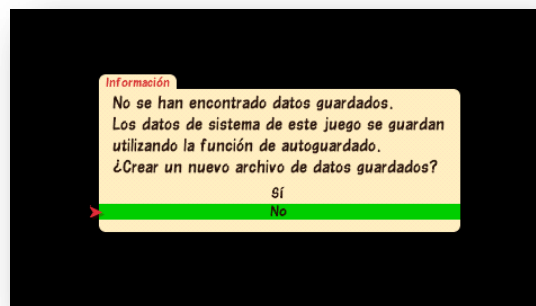


Figura 15: Captura de pantalla del aviso en el videojuego "Patapon".

Los menús principales casi siempre poseen los mismos ítems, como se puede ver en la **Figura 16** y la **Figura 17**:

- New Game / Partida nueva
- Load Game / Continuar
- Extra / Idioma / etc.



Figura 16: Captura de pantalla de menú principal de "Patapon".



Figura 17: Captura de pantalla de menú principal de "the 3rd Birthday".

La mayoría de los juegos disponibles están en 3d. Para estos tipos de juegos es importante tener la zona activa en toda la pantalla o pasiva (menú), como se puede ver en la **Figura 18** y la **Figura 19**.



Figura 18: Ejemplo de la zona "activa" en "the 3rd Birthday".



Figura 19: Ejemplo de la zona "pasiva" en "the 3rd Birthday".

1.3 Logotipos

Al principio fue planificado tener contacto con SCEE para vender el diccionario, en Sony Networks como una aplicación "Minis". Pero como se ha comentado en la introducción, no fue posible contactar con SCEE. Antes de empezar a crear el logotipo, fue necesario buscar las informaciones correspondientes sobre las características de un logotipo.

A través de la **Error! Reference source not found.**, se tiene la posibilidad de ver las características de un logotipo. La tabla siguiente ilustra los logotipos generales.








Tipo	Color	Blanco y negro, monocolor	Color
	Color	Color	Color
Símbolos			
Letras			
Mezcla (Símbolos y letra)			

Tabla 1: Comparación de los logotipos.

Similar a la **Error! Reference source not found.**, la **Tabla 2** ilustra los logotipos de los videojuegos.

Tipo	Color	Blanco y negro, monocolor	Color
	Color	Color	Color
Letras			

³ Logotipo de „Arabia Investing”: http://arabiainvesting.com/About_Us.html


Mezcla (Símbolos y letra)		
---------------------------	---	---

Tabla 2: Comparación de los logotipos de videojuegos.

Creación de logotipo:

Había un logotipo con mis iniciales, fue necesario cambiarlo, porque el estilo de la letra japonesa no resulta muy profesional. Adicionalmente el logotipo debería representar la personalidad de la “empresa” o del “creador”, en este caso mi nombre. En la **Figura 20**, se muestra el primer logotipo.



Figura 20: Primero logotipo.

Antes de cambiar el logotipo original se ha realizado un boceto, como se puede ver en la **Figura 21**.



Figura 21: Boceto de logotipo nuevo.

A través de la **Tabla 3**, se presenta el resultado final de la creación y las pruebas de colores que se han ido haciendo.





	Ejemplo 1	Ejemplo 2
Color 1		
Color 2		

Tabla 3: Resultado final de propio logotipo.

El diseño de logotipo del estudio / proyecto, como se puede ver en la **Figura 22**.



Figura 22: Logotipo del diccionario.

1.4 Mascota

El diccionario debe ser para niños, por este motivo se ha desarrollado una mascota. Todas las aplicaciones para la NDS poseen una mascota o un personaje el cual ayuda al usuario, éste debe dar consejos al usuario en cualquier momento o después de un evento especial. Igualmente es importante que la mascota haga pequeñas animaciones de forma aleatoria en cada intervalo definido para evitar un aburrimiento del usuario. El diseño debe ser muy sencillo, porque los niños no se fijan mucho en los detalles.

Antes de empezar con el diseño de mascota fue necesario hacer un par de análisis y pruebas:

1. ¿Qué mascotas y/o personajes hay en las aplicaciones de aprendizaje y/o diccionario?
2. ¿Es un animal o humano?
3. Si es un animal
 - 3.1 ¿Qué animal es?
 - 3.2 ¿Qué raza corresponde?
 - 3.3 ¿Qué características tiene?
4. Si es un humano
 - 4.1 ¿Qué sexo?
 - 4.2 ¿Cuántos años tiene?
 - 4.3 ¿Qué profesión tiene?
5. Estilo de diseño de carácter ¿manga (japonés) o cómic (americano)?
6. Buscar ejemplos (fotos, dibujos, model sheets, etc.)
7. Hacer varias pruebas

1. ¿Qué mascotas y/o personajes hay en las aplicaciones de aprendizaje y/o diccionario?

Los siguientes AA fueron revisados por el diseño de mascotas:

- Addy – Do you speak English
- Mein Wortschatz-Coach English
- English Buddy

Los demás no tenían mascotas o eran humanos como en “Mein Wortschatz-Coach” y “English Buddz”. Sin embargo en la AA “Addy – Do you speak English” hay dos mascotas, un humano y un animal o alíen (**Figura 23**).



Figura 23: Ejemplo de los mascotas en las AA de NDS.

3. Si es un animal

En este diccionario se ha creado una mascota que es de tipo animal, ya que es un búho de raza "white-faced-owl". Es una raza de búhos pequeños que viven en África (sur y norte).

Las características de los búhos son:

- Muy inteligentes
- Símbolo de sapiencia
- Guardia (Búho de Athenea)

5. Estilo de diseño de carácter ¿manga (japonés) o cómic (americano)?

El estilo de dibujos de la mascota es americano, ya que casi todas las AA están en este estilo en vez de japonés y aunque no hay mucha diferencia entre ambos estilos, esta es notable.

6. Buscar ejemplos (fotos, dibujos, model sheets, etc.)

Como fue comentado en la parte 3, la raza de la mascota será "White-faced-owl". En google, youtube y cualquier página web, se puede encontrar muchas referencias a esta raza de búhos. Por ejemplo: la **Figura 24**, **Figura 25** y la **Figura 26**.

Se tiene que tener en cuenta que en este estudio la mascota estará dibujada. Entonces también fue importante encontrar ejemplos de búhos en la animación y Walt Disney tiene dos búhos muy conocidos y famosos.

- **Arquimides** de la película de "Merlín el encantado"
- **Búho** de la serie y película de "Winnie Pooh"



Figura 24: varias fotos de "white-faced-owl" de la búsqueda de google.



Figura 25: Foto de búho: "white-faced-owl". Referencia Youtube: <http://www.youtube.com/watch?v=fRSCI1HWBC4>.

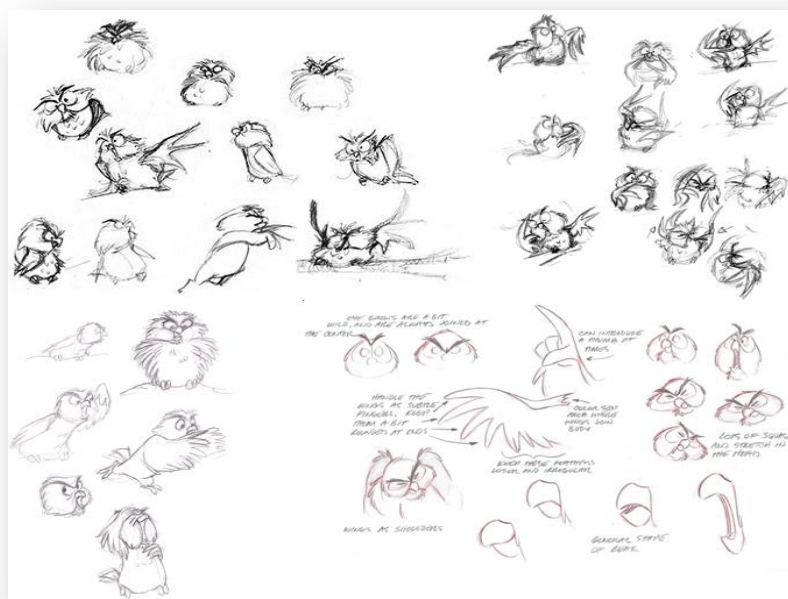


Figura 26: Model Sheet de búhos de Disney.

7 Hacer varias pruebas

Sin saber de que raza corresponde “Profesor Tobí”, se ha hecho unos dibujos. Al principio, el búho debería llevar un gorro de graduación para mostrar su inteligencia. Se muestra unos dibujos en la **Figura 27**.



Figura 27: dibujos principales del búho.

En los dibujos principales, el búho tenía dos plumas como cejas. Sin embargo dichas dos plumas son en realidad como orejas. Algunos búhos tienen estas plumas como orejas. En la **Figura 28** se puede ver el cambio.



Figura 28: Cambio de pluma "ceja" a pluma "oreja".

Después de analizar unos detalles sobre los búhos fue obstante realizar un “esqueleto” cómo se dibuja el carácter “Profesor Tobí”. Dicho “esqueleto” es necesario también para poder realizar animaciones en 2D o en 3D.

La **Figura 29**, muestra el “esqueleto” del “Tobí”.

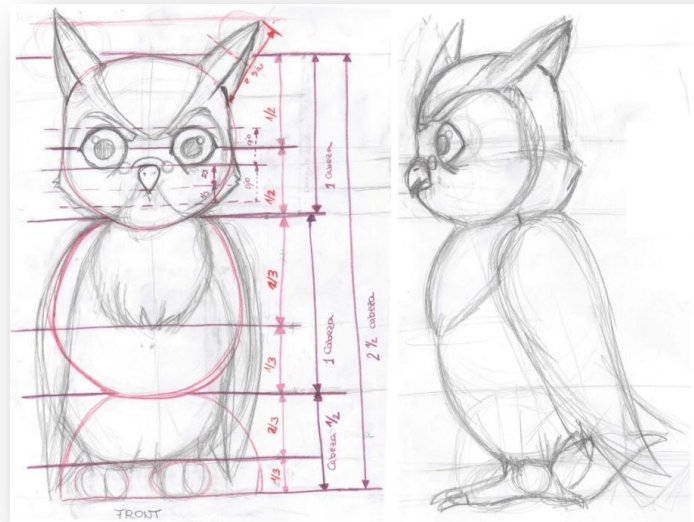


Figura 29: Esqueleton del "Profesor Tobí".

Con el “eskeleton” es posible hacer unas pruebas de gestion de emociones. Otra parte muy importante en el tema de animación. En la **Figura 30**, se muestra unos ejemplos de emociones que puede hacer “Tobí”.



Figura 30: Emociones del "Profesor Tobí".

Hay que hacer unas pruebas de color. Sabiendo la raza, fue obstarnte comproar los colores para tener un resultado final, como se puerder ver en la **Figura 31**.



Figura 31: De boceto a color.

Por última se realiza una animación simple. En el diccionario la animación consiste en el movimiendo de los ojos. La **Figura 32** muestra una pista de animación del búho.



Figura 32: Animación del búho.

2. Información adicional

En esta parte se enseña cómo sacar el OSK original de la PSP y cómo añadir música en su propio XMB en la PSP. Para añadir audio en el XMB es obstatante instalarse un códec para el formato de Atrac.

Entonces los requisitos son los siguientes:

- Descargar Atrac "Converter"
- Descargar Codwave
- Descargar GWAT

Después de instalar todo hay que seleccionar una pista de audio. En Godwave se debe abrir el fichero y guardarlo con el nombre "SDN0" con el tipo de fichero de "wav" con los atributos "Atrac3 66kbps stereo" (el tecerco de abajo). Si no se puede encontrar el atributo Atrac3, entonces hay que volver a instalar el Atrac "Converter". Ahora es importante cambiar la extensión del fichero "SDN0.wav" a "SDN0.at3". Ya se puede usarlo como un fichero de sonido en el Makefile para el EBOOT.PBP

Si se quiere reproducir el sonido en un bucle infinito, entonces hay que seguir con la modificación del audio. Ahora se necesita el GWAT para crear el bucle. Abrir el GWAT y copiar el fichero dentro del GWAT a través de "Drag and Drop" y seleccionar Godwave. Va a guardarlo con un bucle. Después de guardar se encuentra un fichero "looped- SDN0.at3". Borrar "SDN0.at3" y cambiar el nombre de "looped- SDN0.at3" a "SDN0.at3". Ahora hay que escribir en el Makefile: **PSP_EBOOT_SND0= SND0.at3**.



OJO: Es la pista de audio no debería ser más de 30 segundos, si no la PSP reproduce el audio más rápido que debería.

Como ya comentado, es posible usar el OSK original de la PSP. El ejemplo está dado en la SDK MinPSPw.

El código para mostrar y usar el OSDK:

```
#include <pspkernel.h>
#include <pspdisplay.h>
#include <pspdebug.h>
#include <pspgu.h>
#include <string.h>
#include <psputility.h>

PSP_MODULE_INFO("OSK Sample", 0, 1, 1);
PSP_MAIN_THREAD_ATTR(THREAD_ATTR_USER);

static int done = 0;

int exit_callback(int arg1, int arg2, void *common)
{
    done = 1;

    return 0;
}

int CallbackThread(SceSize args, void *argp)
{
    int cbid = sceKernelCreateCallback("Exit Callback", exit_callback, NULL);
    sceKernelRegisterExitCallback(cbid);
    sceKernelSleepThreadCB();

    return 0;
}

int SetupCallbacks(void)
{
    int thid = sceKernelCreateThread("update_thread", CallbackThread, 0x11, 0xFA0,
    PSP_THREAD_ATTR_USER, 0);

    if(thid >= 0)
        sceKernelStartThread(thid, 0, 0);

    return thid;
}

static unsigned int __attribute__((aligned(16))) list[262144];

#define BUF_WIDTH      (512)
#define SCR_WIDTH      (480)
#define SCR_HEIGHT     (272)
#define PIXEL_SIZE(4)
#define FRAME_SIZE     (BUF_WIDTH * SCR_HEIGHT * PIXEL_SIZE)
#define ZBUF_SIZE (BUF_WIDTH SCR_HEIGHT * 2)

#define NUM_INPUT_FIELDS(3)
#define TEXT_LENGTH          (128)

int main(int argc, char* argv[])
{
    SetupCallbacks();

    sceGuInit();
    sceGuStart(GU_DIRECT, list);
    sceGuDrawBuffer(GU_PSM_8888, (void*)0, BUF_WIDTH);
```

```

sceGuDispBuffer(SCR_WIDTH,SCR_HEIGHT,(void*)0x88000,BUF_WIDTH);
sceGuDepthBuffer((void*)0x110000,BUF_WIDTH);
sceGuOffset(2048 - (SCR_WIDTH/2),2048 - (SCR_HEIGHT/2));
sceGuViewport(2048,2048,SCR_WIDTH,SCR_HEIGHT);
sceGuDepthRange(0xc350,0x2710);
sceGuScissor(0,0,SCR_WIDTH,SCR_HEIGHT);
sceGuEnable(GU_SCISSOR_TEST);
sceGuDepthFunc(GU_EQUAL);
sceGuEnable(GU_DEPTH_TEST);
sceGuFrontFace(GU_CW);
sceGuShadeModel(GU_FLAT);
sceGuEnable(GU_CULL_FACE);
sceGuEnable(GU_TEXTURE_2D);
sceGuEnable(GU_CLIP_PLANES);
sceGuFinish();
sceGuSync(0,0);
sceDisplayWaitVblankStart();
sceGuDisplay(GU_TRUE);

unsigned short intext[NUM_INPUT_FIELDS][TEXT_LENGTH];
unsigned short outtext[NUM_INPUT_FIELDS][TEXT_LENGTH];
unsigned short desc[NUM_INPUT_FIELDS][TEXT_LENGTH];

memset(&intext, 0, NUM_INPUT_FIELDS * TEXT_LENGTH * sizeof(unsigned short));
memset(&outtext, 0, NUM_INPUT_FIELDS * TEXT_LENGTH * sizeof(unsigned short));
memset(&desc, 0, NUM_INPUT_FIELDS * TEXT_LENGTH * sizeof(unsigned short));

int i;

for(i = 0; i < NUM_INPUT_FIELDS; i++)
{
    desc[i][0] = 'F';
    desc[i][1] = 'i';
    desc[i][2] = 'e';
    desc[i][3] = 'l';
    desc[i][4] = 'd';
    desc[i][5] = ' ';
    desc[i][6] = i + 48 + 1; // Convert i to ASCII value.
    desc[i][7] = 0;

    intext[i][0] = 'T';
    intext[i][1] = 'e';
    intext[i][2] = 'x';
    intext[i][3] = 't';
    intext[i][4] = ' ';
    intext[i][5] = i + 48 + 1; // Convert i to ASCII value.
    intext[i][6] = 0;

}

SceUtilityOskData data[NUM_INPUT_FIELDS];

for(i = 0; i < NUM_INPUT_FIELDS; i++)
{
    memset(&data[i], 0, sizeof(SceUtilityOskData));
    data[i].language = PSP_UTILITY_OSK_LANGUAGE_DEFAULT; // Use system default for
text input
    data[i].lines = 1;
    data[i].unk_24 = 1;
    data[i].inputtype = PSP_UTILITY_OSK_INPUTTYPE_ALL; // Allow all input types
    data[i].desc = desc[i];
    data[i].intext = intext[i];
    data[i].outtextlength = TEXT_LENGTH;
    data[i].outtextlimit = 32; // Limit input to 32 characters

```

```

        data[i].outtext = outtext[i];
    }

    SceUtilityOskParams params;
    memset(&params, 0, sizeof(params));
    params.base.size = sizeof(params);
    sceUtilityGetSystemParamInt(PSP_SYSTEMPARAM_ID_INT_LANGUAGE,
    &params.base.language);
    sceUtilityGetSystemParamInt(PSP_SYSTEMPARAM_ID_INT_UNKNOWN,
    &params.base.buttonSwap);
    params.base.graphicsThread = 17;
    params.base.accessThread = 19;
    params.base.fontThread = 18;
    params.base.soundThread = 16;
    params.datacount = NUM_INPUT_FIELDS;
    params.data = data;

    sceUtilityOskInitStart(&params);

    while(!done)
    {
        sceGuStart(GU_DIRECT, list);
        sceGuClearColor(0);
        sceGuClearDepth(0);
        sceGuClear(GU_COLOR_BUFFER_BIT|GU_DEPTH_BUFFER_BIT);

        sceGuFinish();
        sceGuSync(0,0);

        switch(sceUtilityOskGetStatus())
        {
            case PSP_UTILITY_DIALOG_INIT:
                break;

            case PSP_UTILITY_DIALOG_VISIBLE:
                sceUtilityOskUpdate(1);
                break;

            case PSP_UTILITY_DIALOG_QUIT:
                sceUtilityOskShutdownStart();
                break;

            case PSP_UTILITY_DIALOG_FINISHED:
                break;

            case PSP_UTILITY_DIALOG_NONE:
                done = 1;

            default :
                break;
        }

        sceDisplayWaitVblankStart();
        sceGuSwapBuffers();
    }

    pspDebugScreenInit();
    pspDebugScreenSetXY(0, 0);

    int j;

    for(i = 0; i < NUM_INPUT_FIELDS; i++)
    {
        pspDebugScreenPrintf("Field %d: ", i+1);
    }

```

```

switch(data[i].result)
{
    case PSP_UTILITY_OSK_RESULT_UNCHANGED:
        pspDebugScreenPrintf("UNCHANGED: ");
        break;

    case PSP_UTILITY_OSK_RESULT_CANCELLED:
        pspDebugScreenPrintf("CANCELLED: ");
        break;

    case PSP_UTILITY_OSK_RESULT_CHANGED:
        pspDebugScreenPrintf("CHANGED: ");
        break;

    default:
        break;
}

for(j = 0; data[i].outtext[j]; j++)
{
    unsigned c = data[i].outtext[j];

    if(32 <= c && c <= 127) // print ascii only
        pspDebugScreenPrintf("%c", data[i].outtext[j]);
}

    pspDebugScreenPrintf("\n");
}

done = 0;

while(!done)
    sceKernelDelayThread(1000);

sceGuTerm();
sceKernelExitGame();
return 0;
}

```

Y su Makefile:

```

TARGET = osk
OBS = main.o
CFLAGS = -O2 -G0 -Wall
CXXFLAGS = $(CFLAGS) -fno-exceptions -fno-rtti
ASFLAGS = $(CFLAGS)
BUILD_PRX = 1
LIBS = -lpspgu -lpsputility
EXTRA_TARGETS = EBOOT.PBP
PSP_EBOOT_TITLE = PSP OSK Sample
PSPSDK=$(shell psp-config --pspsdk-path)
include $(PSPSDK)/lib/build.mak

```

Bibliografía

Sonido en el Eboot. <http://www.devils-shadow.com/forums/psp-homebrew-hacks-apps-games/2969-psp-eboot-sounds.html>, llamada 2010-12-04

Declaración jurada

Yo afirmo que este estudio mío lo realicé totalmente por mi cuenta sin apoyo ilegal. Marqué todos los enfoques usados que no son míos con citas directas o indirectas. Estoy segura en que cada aseguración mala tiene consecuencias jurídicas.

Todas las imágenes utilizadas en este estudio, que no fueron creadas por mí son con derechos de autor. Yo solo he realizado las capturas de pantalla y ninguna modificación.

Valencia, 30 de Septiembre de 2011

Katharina Maria Kaczmarczyk