

Capítulo 4: Análisis y migración de un Tutor Interactivo

4.1 Análisis de la estructura y funcionamiento del FourSquare

En este capítulo se utilizará el tutor interactivo FourSquare en español como ejemplo para describir el funcionamiento de los tutores y las tablas de la base de datos que son utilizadas por éstos. También se explicarán los cambios realizados a los códigos para que funcionara con la base de datos ya migrada a MySQL.

En la Figura 4.1.1 se muestra la interfaz del tutor interactivo, la cual esta compuesta por cuatro botones en donde se mostrarán las cuatro palabras de la actividad. En la parte izquierda se muestra el agente animado que se encargará de pronunciar las palabras de la actividad y de retroalimentar al alumno; debajo de éste se encuentra un boton “again” que el usuario puede presionar si desea volver a oír la palabra pronunciada por el agente. En la parte inferior derecha se muestra una barra con los niveles disponibles para la actividad y por último, en azul, en la parte izquierda se encuentra un panel azul en donde se muestran las monedas ganadas por el alumno.

FourSquare funciona de la siguiente manera: el tutor presenta al alumno cuatro palabras diferentes en la pantalla, dependiendo del nivel escogido por el usuario, a través del agente animado le pide que presione el cuadro de la palabra que pronunciará. Si el alumno presiona la palabra correcta en el primer intento, entonces el tutor le dará una moneda de oro como recompensa, si lo hace al segundo intento entonces le da como recompensa una moneda de plata, y si no lo logra entonces le da una moneda de bronce junto con la respuesta correcta.

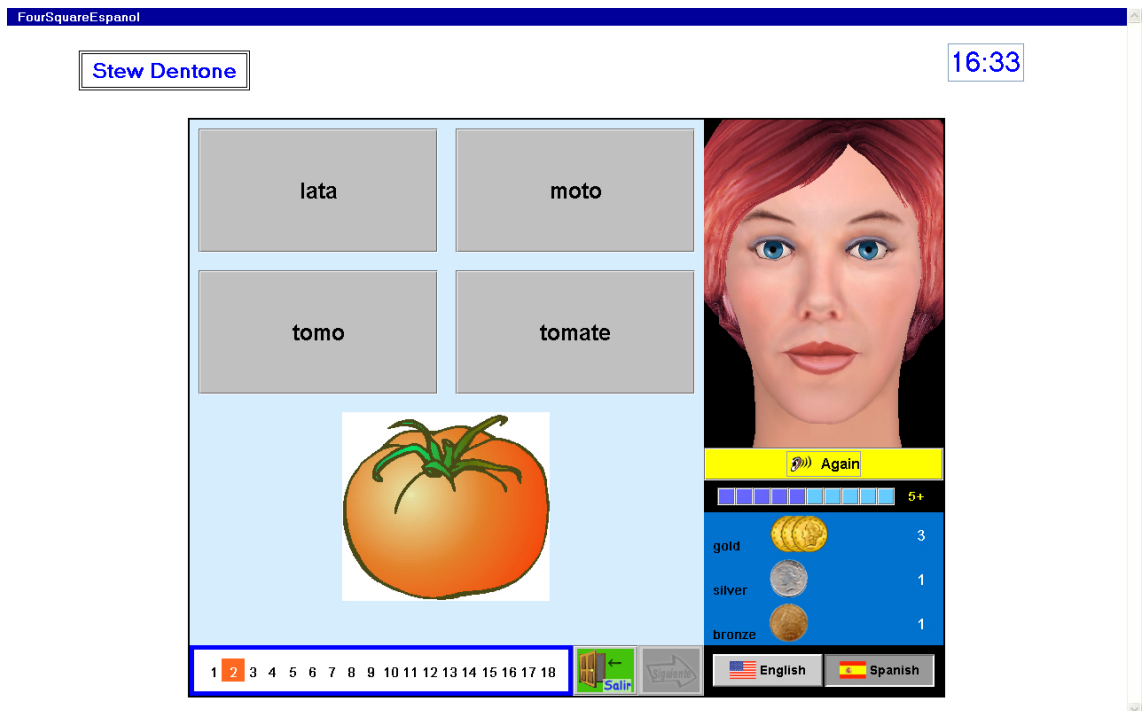


Figura 4.1.1 Interfaz del Tutor interactivo FourSquare

El usuario al acceder al tutor tiene tres opciones mostradas en la Figura 4.1.2: la primera opción es la de empezar la actividad, cuando el usuario elige esta opción, el tutor se conecta a la base de datos para recuperar la serie de palabras que utilizará en esa actividad, busca las imágenes y se conecta al servidor de voz mediante el cual el agente animado le dará retroalimentación al usuario.

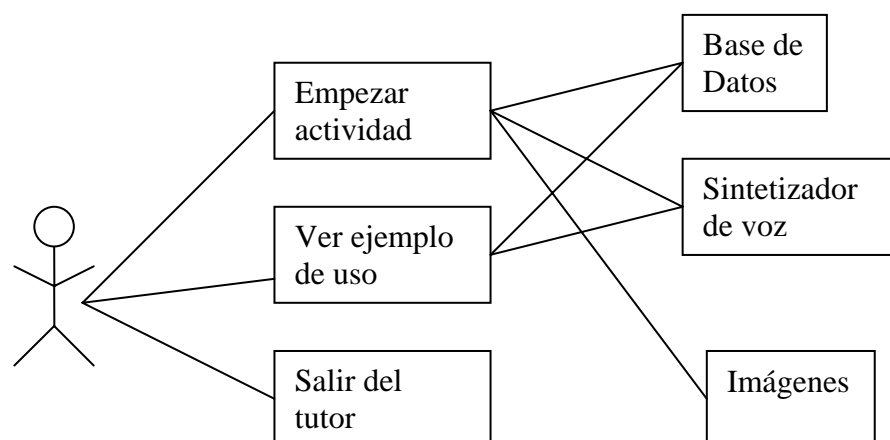


Figura 4.1.2 Caso de uso del tutor FourSquare

La segunda opción es la de ver ejemplo de uso, en la cual el tutor le muestra al usuario un ejemplo de cómo funciona el juego. Para esto el tutor accesa a la base de datos para recuperar las palabras que usará para el ejemplo y al servidor de voz para que el agente animado pueda hablar.

La tercera opción es la de salir del tutor, con la cual se cierra.

Los datos con los que opera el tutor, son obtenidos con la ayuda del CSLR.jar el cual contiene el código encargado de acceder a la base de datos.

La lista de palabras que presentará al usuario en cada actividad se encuentran originalmente en la tabla words_four_square.

Tabla words_four_Square	
fq_id	Int (11)
fq_word	varchar (30)
fq_foils	text
fq_word_phoneme	varchar(60)
fq_word_phoneme_cvt	varchar(20)

Tabla 4.1.3 Estructura de la tabla words_four_square de la base de datos

Esta tabla esta compuesta por cinco campos:

fq_id: un entero que nos indica el nivel al que pertenece esa palabra.

fq_word: guarda la palabra que el tutor le pide al alumno encontrar entre las cuatro posibles opciones.

fq_foils: un conjunto de cuatro palabras, de entre las cuales se seleccionarán tres, para presentar como posibles opciones de respuesta del ejercicio.

fq_word_phoneme guarda la descripción fonética de la palabra; y fq_word_phoneme_cvt guarda la descripción consonante-vocal de la palabra solicitada.

Sin embargo la base de datos tiene creadas vistas a partir de dicha tabla y de acuerdo al nivel al que pertenecen las palabras así como la forma en que están compuestas fonéticamente; esto es con el propósito de simplificar la búsqueda de las palabras. Son éstas vistas las que FourSquare accesa para la recuperación de las palabras.

Una vista se encuentra estructurada de la siguiente manera:

Vista words_level_502	
cmu_word	Text
fq_id	int(11)
fq_word	varchar(30)
fq_foils	Text
fq_word_phoneme	varchar(60)
query_type	varchar(1)

Tabla 4.1.4 Estructura de la vista words_level_502 de la base de datos

Todas las vistas se encuentran estructuradas de acuerdo a lo mostrado en la Tabla 4.1.4: en cmu_word se concatenan los datos de fq_id y fq_word de la tabla words_four_square; selecciona los campos fq_id y fq_word, fq_word_phoneme y asigna un valor de 0 al nuevo campo query_type.

El acceso a las vistas lo hace mediante el programa JTutorLevel que se encuentra en el directorio JTutorUtil del archivo CSLR.jar; a continuación se muestra el

fragmento de código en el cual se recuperan las palabras buscándolas en la vista que guarda las palabras correspondientes al nivel seleccionado para la actividad.

```

/*****
    sql = " SELECT * FROM WORDS_LEVEL_" + LEVEL[level];
    if(!sql.equals(""))
        try
        {
            PreparedStatement ps = db.prepareStatement(sql);
            ResultSet rs = ps.executeQuery();
            int pick = 0;
            while(rs.next())
                if(wordList.size() > 0)
                {
                    int pick2 = pick = randomStatic.nextInt(wordList.size());
                    wordList.insertElementAt(new JTutorWord(rs.getString("fq_word"),
                        rs.getString("fq_foils")), pick2);
                } else
                {
                    wordList.add(new JTutorWord(rs.getString("fq_word"), rs.getString("fq_foils")));
                }
            }
        } catch(Exception e)
        {
            System.out.println("ERROR " + e);
        }
}
*****/

```

La instrucción `sql = " SELECT * FROM WORDS_LEVEL_" + LEVEL[level];` selecciona la vista que guarda las palabras del nivel buscado; y obtiene la palabra correcta que está en `fq_word` y las palabras opcionales que se encuentran en `fq_foils`.

Estas palabras son finalmente regresadas a FourSquare el cual las almacena en un arreglo.

Los sonidos de las palabras y de las frases que utiliza para retroalimentar al alumno, los obtiene de las siguientes tablas:

phoneme_sound_default

word_sound_default

word_sound_mx

nword_sound_default

nword_sound_default_mx

tutor_sound_default

tutor_sound_mx

Todas las tablas anteriores cuentan con la misma estructura; y para mostrarla se toma como ejemplo la tabla tutor_sound_mx:

Tabla tutor_sound_mx	
s_string	text
s_m_stream	longblob
s_f_stream	longblob
s_time	int(11)

Tabla 4.1.4 Estructura de la tabla tutor_sound_mx de la base de datos

Esta tabla contiene los campos s_string que guarda el texto del sonido grabado; s_m_stream y s_f_stream guarda el sonido en forma de bytes; s_time guarda la duración de sonido.

El proceso para la recuperación de los sonidos se muestra en el siguiente diagrama:

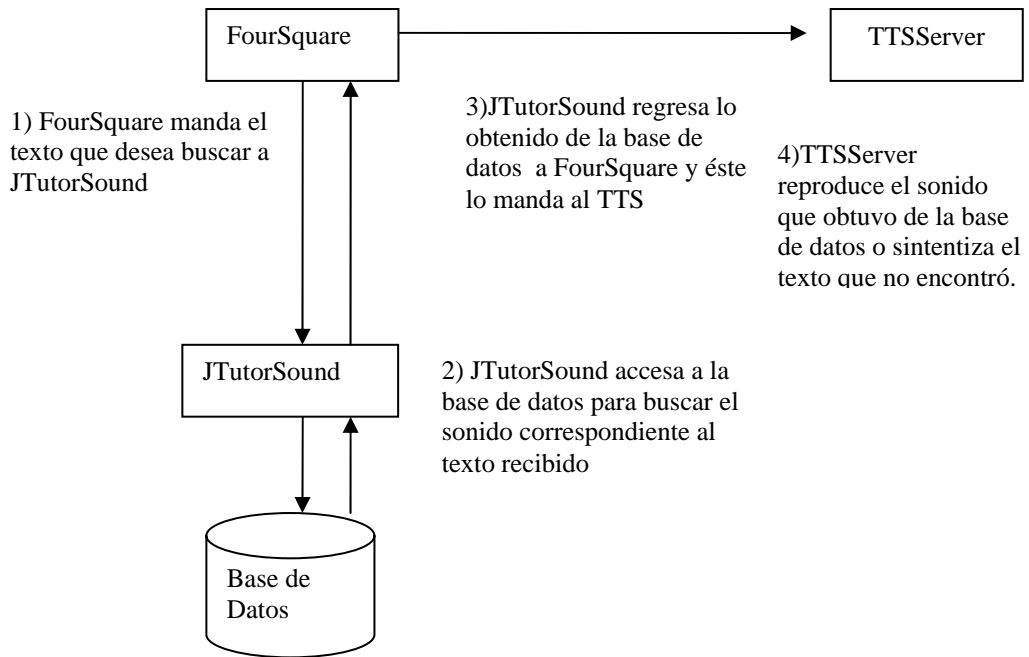


Figura 4.1.4 Diagrama que muestra como Foursquare recupera los sonidos de la base de datos

En FourSquare la función llamada SendText manda a getJCSLRSOBStream, función de la clase JTutorSound, el texto del cual desea recuperar su sonido.

getJCSLRSOBStream con la ayuda de las funciones getDefaultStreamByWord, getMXStreamBySentence, getMXDefaultStreamByWord, getDefaultStreamBySentence y getDefaultStreamByNoneWord, realiza una consulta a la base de datos para buscar el sonido correspondiente al texto recibido. Si encuentra el sonido entonces regresa un tipo de dato JCSLRSOBStream. SendText recibe este tipo de dato y reproduce el sonido a través del método SendStream(JCSLRSOBStream) de la clase JTTS; si JTutorSound no encuentra el sonido en la base de datos, entonces regresa null a FourSquare éste manda el texto al TTSServer el cual lo sintetizará para generar el sonido correspondiente.

```

/*****/

private void sendText(String text,String language)
{
    try
    {
        JCSLRSOBStream sob =
        JTutorSound.getJCCLRStream(db,text,false,language);
        if(sob != null)
        {
            jtts.sendStream(sob);
        }
        else
        {
            jtts.sendText(text);
        }
    }
}

/*****/

```

4.2 Cambios al código fuente del FourSquare y los archivos involucrados en su funcionamiento.

Una vez que se explicó el funcionamiento y estructura del tutor FourSquare en la Figura 4.2.1 podemos ver que los que están involucrados con su funcionamiento son JFourSquare.jar, con los códigos del tutor, CSLR.jar con los códigos encargados de la conexión a la base de datos, recuperación de sonidos y manejo de datos, y JFourSquare.xml, con los comandos para la retroalimentación al alumno y el archivo JFourSquare.php

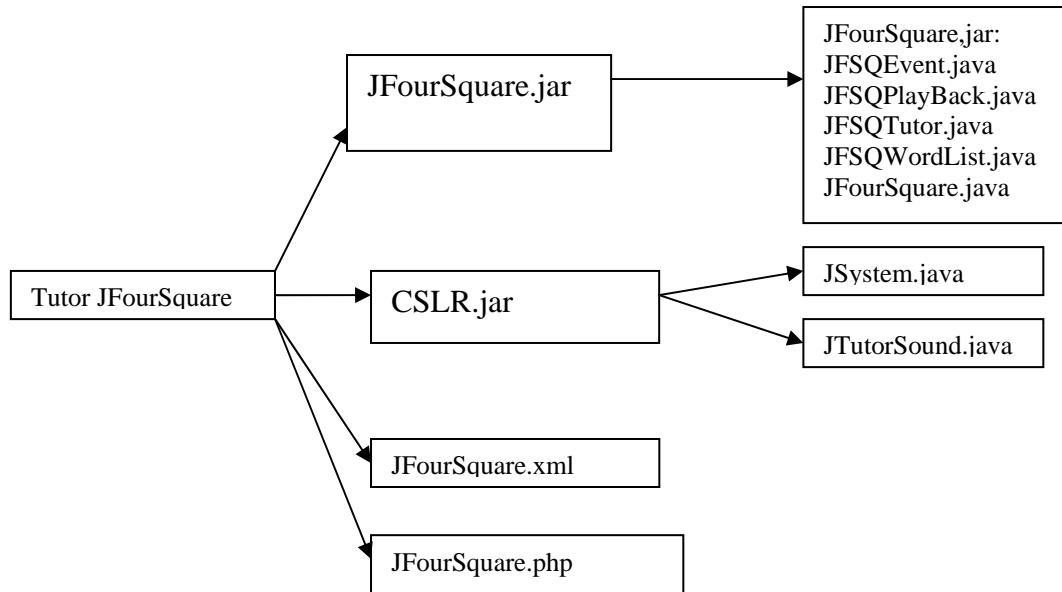


Figura 4.2.1 Archivos involucrados en el funcionamiento del FourSquare

En este caso en particular para poder lograr que el tutor FourSquare funcionara con la base de datos de MySQL fueron realizados los cambios a los archivos JSystem.java del CSLR.jar para que realizara la conexión a la base de datos de MySQL, modificación que se explica en la sección 3.4 “Conexión del sistema con MySQL para la autenticación del usuario”. Se realizaron cambios al archivo JTutorSound para que buscara los sonidos en MySQL sin hacer uso de su OID (Object Identifier), procedimiento que se explica en la sección 3.3 Conexión de los tutores, libros interactivos y Festival con MySQL. En cuanto al archivo JFourSquare.xml fue necesario realizar cambios a las instrucciones que iba a buscar en la base de datos ya que no coincidían todas con las almacenadas en la base de datos; y el archivo JFourSquare.php en el cual se tuvieron que modificar las rutas donde las imágenes que el tutor utiliza se encuentran almacenadas.

Es necesario aclarar que no todos los tutores utilizan a JSystem.java para hacer la conexión a la base de datos, algunos lo hacen por su propia cuenta dentro de su código, así en esos casos si es necesario realizar los cambios correspondientes a la conexión de la base de datos dentro del código del tutor.

4.3 Pruebas

Aquí se muestran las pruebas al funcionamiento de los tutores ya trabajando con la base de datos en MySQL. La técnica utilizada para realizar las pruebas fue la de caja negra. Las actividades evaluadas fueron las siguientes:

- A.- Iniciar actividad al presionar “Empezar” o “Stara”.
- B.- Mostrar ejemplo de funcionamiento al presionar “Demo” o Ejemplo”
- C.- Salir del tutor al presionar el boton “Salir”
- D.-Recuperación de las imágenes.
- E.- Recuperación de los sonidos
- F.- Recuperación de las palabras
- G.- La tarea funcionó de manera correcta.

En la evaluación se dio una “s” si la actividad evaluada la realizó y una “n” si no fue realizada por el tutor , una “p” si la realiza de manera parcial y un “n/a” (no aplica) en caso de que dicha actividad nunca fue realizada por el tutor antes de realizar la migración.

	ACTIVIDAD						
TUTOR	A	B	C	D	E	F	G
FourSquare	s	s	s	s	s	s	s
Matching	s	s	s	s	s	s	s
Matching Letter - Letter sound,	s	s	s	s	s	s	s
Matching Lower - Lower	s	s	s	s	s	s	s
Matching Lower - Upper	s	s	s	s	s	s	s
Matching Picture - Letter	s	s	s	s	s	s	s
Matching Picture - Letter (blends)	s	s	s	s	s	s	s
Matching Picture-Letter (digraphs)	s	s	s	s	s	s	s
Matching Picture - Picture	s	s	s	s	s	s	s
Matching Picture - Word	s	s	s	s	s	s	s
Matching Upper - Upper	s	s	s	s	s	s	s
Matching Word - Letter	s	s	s	s	s	s	s
Matching Word - Letter sound	s	s	s	s	s	s	s
Matching Word - Letter sound (blends)	s	s	s	s	s	s	s
Matching Word - Word (different font)	s	s	s	s	s	s	s
Matching Word - Word (same font)	s	s	s	s	s	s	s
Beginning Sounds	s	s	s	s	s	s	s
Beginning Sounds (No Letter)	s	s	s	s	s	s	s
Ending Sounds	s	s	s	s	s	s	s

Making Words	s	s	s	s	s	s	s
--------------	---	---	---	---	---	---	---

Figura 4.3.1 Tabla de pruebas de funcionamiento de los tutores del CSLR

En la siguiente tabla se engloban en un solo grupo los tutores no modificados y en otro grupo, los libros interactivos no modificados.

	ACTIVIDAD						
TUTOR/LIBRO	A	B	C	D	E	F	G
Tutores_no_modificados	n	n	s	p	n	n	n
Libros_no_modificados	n	n	s	p	n	n	n

Figura 4.3.2 Tabla de pruebas de funcionamiento de los tutores no modificados del CSLR

La búsqueda y recuperación de los sonidos por parte de los tutores aunque funcionó fue lenta. Se pudo observar que el tutor si estaba obteniendo los datos correctamente de la base de datos, sin embargo no todos los sonidos se encontraban adentro de la base de datos, por lo que el tutor hacía uso del sintetizador de voz para poder crear esos sonidos; lo que alenta más el funcionamiento del tutor.