

**PROTOTIPO DE APLICACIÓN WEB PARA CREACIÓN Y
ADMINISTRACIÓN DE APLICACIONES DE VOZ ELABORADAS
CON VXML PARA APRENDIZAJE MÓVIL**



**TESIS
MAESTRÍA EN CIENCIAS EN TECNOLOGÍA INFORMÁTICA**

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS
SUPERIORES DE MONTERREY**

**POR
ING. CÉSAR CASIMIRO TOLEDANO LUNA**

MAYO DE 2011

**Prototipo de Aplicación web para creación y
administración de aplicaciones de voz elaboradas con
VXML para aprendizaje móvil**

TESIS

**Maestría en Ciencias en
Tecnología Informática**

Instituto Tecnológico y de Estudios Superiores de Monterrey

Por

Ing. César Casimiro Toledano Luna

Mayo de 2011

**Prototipo de Aplicación web para creación y
administración de aplicaciones de voz elaboradas con
VXML para aprendizaje móvil**

Por

Ing. César Casimiro Toledano Luna



TESIS

**Presentada a la División de Mecatrónica y Tecnologías de Información
Este trabajo es requisito parcial para obtener el grado académico de
Maestro en Ciencias en Tecnología Informática**

**Instituto Tecnológico y de Estudios Superiores de
Monterrey**

Monterrey, N.L., Mayo de 2011

A mi familia y novia por estar siempre al pendiente no solo de este trabajo sino de todos los aspectos importantes de mi vida.

Reconocimientos

A mi asesor, el Dr. Guillermo Jiménez Pérez, por ser parte central en la guía, seguimiento y motivación para la realización de este trabajo.

A los sinodales el Dr. José Icaza y el Dr. Carlos Astengo por su interés y ayuda en la mejora de este trabajo.

Al Dr. Juan Arturo Nolazco por proveerme de un servidor para pruebas del Departamento de Ciencias Computacionales y por su ayuda para agilizar los trámites de autorización necesarios del Departamento de Redes.

Al Ing. Roberto Aceves por brindarme su tiempo asesorándome en la configuración del servidor.

Cesar Casimiro Toledano Luna

*Instituto Tecnológico y de Estudios Superiores de Monterrey
Mayo 2011*

Prototipo de Aplicación web para creación y administración de aplicaciones de voz elaboradas con VXML para aprendizaje móvil

Cesar Casimiro Toledano Luna, M.C.
Instituto Tecnológico y de Estudios Superiores de Monterrey

Asesor de la tesis: Dr. Guillermo Jiménez Pérez

El aprendizaje móvil brinda a los alumnos de esta modalidad educativa de la capacidad de adquirir conocimientos a cualquier hora en cualquier lugar. Lo anterior es logrado gracias a las funcionalidades de los dispositivos móviles utilizados, tales como el envío y recepción de mensajes SMS, el acceso a internet, etc. Entre los dispositivos móviles utilizados en el aprendizaje móvil se pueden encontrar los teléfonos celulares o los Smartphones. Por ser el aprendizaje en el aprendizaje móvil dependiente de las funcionalidades de los dispositivos móviles, los alumnos obtendrán un mayor aprendizaje cuando la tecnología sea aprovechada eficientemente.

A pesar de que existen proyectos de aprendizaje móvil haciendo uso de teléfonos celulares con historias de éxito, aun se pueden detectar algunas desventajas tecnológicas de estos dispositivos para el envío y recepción de información, lo que repercute directamente en la transmisión de conocimientos y la obtención de un aprendizaje. Entre las anteriores se pueden encontrar los pequeños tamaños de pantalla o los métodos de ingreso de datos incómodos. En el caso de los teléfonos celulares, lo anterior puede ser evitado aprovechando la propiedad intrínseca de audio para el envío y recepción de información, de esta forma el dispositivo es usado para lo cual fue construido.

La tarea de transmisión de conocimientos con teléfonos celulares puede realizarse a través de recursos de aprendizaje de audio, los cuales pueden ser representados por aplicaciones de voz, es decir programas computacionales que interactúan con sus usuarios a través de la línea telefónica. Esta interacción es lograda gracias a la tecnología VXML, un lenguaje y estándar del World Wide Web Consortium, que brinda a las aplicaciones de voz con la capacidad de reproducir texto de manera hablada y la capacidad de reconocer la voz como mecanismo de ingreso o manejo de datos. Esta tecnología permite la creación sencilla de aplicaciones de voz de manera muy similar a lo que es la creación de páginas web con el lenguaje HTML, no obstante es necesario tener un conocimiento sobre el lenguaje y sobre otras herramientas tecnológicas para elaboración de las aplicaciones de voz.

En este trabajo se presenta un prototipo de aplicación web desarrollado con tecnologías computacionales como los son las tecnologías Web 2.0 y los marcos de

trabajo de Spring y Hibernate, con el objetivo de mostrar cómo pueden ser creadas y administradas de manera sencilla aplicaciones de voz que consistan en recursos de aprendizaje de audio. La elaboración de los recursos de aprendizaje depende en gran medida de las capacidades que ofrece la tecnología VXML y es por esto que se presentan, como parte también de la contribución de este trabajo, los exámenes múltiples, exámenes orales y blog, para ejemplificar cómo estos pueden ser diseñados. En el caso de los exámenes múltiples, se aprovecha la capacidad de creación de formas y campos para la simulación de preguntas y posibles respuestas. En el caso de exámenes orales se aprovecha la capacidad de grabación en formato de audio de las respuestas orales dadas por los alumnos. Por último el blog aprovecha las capacidades anteriores adicionalmente de la capacidad de reproducción de archivos de audio previamente procesados, como en el caso de los exámenes orales.

Este trabajo se considera como una propuesta de proyecto de aprendizaje móvil y también como una herramienta tecnológica complementaria a las ya establecidas para la administración de recursos de aprendizaje. Siendo la aplicación un prototipo, existen muchas posibilidades de mejora, sin embargo se cumple con el objetivo de mostrar el cómo las tecnologías computacionales contribuyen en facilitar la creación y administración de los recursos de aprendizaje de audio, visualizados como elementos que contribuyen en la entrega de conocimientos por medio de los teléfonos celulares.

Índice General

Reconocimientos	VI
Resumen.....	VII
Índice de Figuras	XIII
Capítulo 1 Planteamientos Generales.....	1
1.1 Situación al problema.....	1
1.2 Planteamiento del problema de investigación	3
1.3 variables del planteamiento del problema de investigación	3
1.4 Objetivos de la investigación	4
1.5 Preguntas de investigación	4
1.6 Justificación de la investigación	5
1.7 Contexto de la investigación	6
1.8 Alcance de la investigación	6
1.9 Resumen del capítulo	7
Capítulo 2 Marco Teórico.....	8
2.1 Aprendizaje móvil.....	9
2.1.1 Definición.....	9
2.1.2 Características.....	9
2.1.3 Modelos e impacto en los alumnos	10
2.1.4 Antecedentes de proyectos de aprendizaje móvil.....	11
2.1.4.1 ABC Mobile Phone.....	12
2.1.4.2 MobileD	13
2.1.4.3 BBC Janala	13
2.1.4.4 MILLEE	14
2.1.4.5 Otros proyectos con uso de teléfonos móviles.....	14
2.1.5 Resumen de sección de aprendizaje móvil	15
2.2 Web 2.0	16
2.2.1 Definición de Web 2.0.....	16
2.2.2 Ejemplos de aplicaciones Web 2.0.....	17

2.2.3	Tecnologías Web 2.0	17
2.2.4	Ajax.....	18
2.2.4.1	Funcionamiento Ajax.....	18
2.2.4.2	Librerías Ajax	19
2.2.5	XSLT.....	20
2.2.6	Funcionamiento XSLT	20
2.2.7	jQuery	22
2.2.8	Funcionamiento jQuery.....	22
2.2.9	Resumen de sección de web 2.0.....	23
2.3	Lenguaje VXML.....	24
2.3.1	Definición.....	24
2.3.2	Infraestructura para despliegue de aplicaciones de voz.....	24
2.3.3	Diseño de aplicaciones de voz.....	25
2.4	Resumen del capítulo	27
Capítulo 3	Propuesta	28
3.1	Consideraciones de la tecnología VXML en el surgimiento de la propuesta	28
3.2	Descripción de la propuesta	28
3.2.1	Utilización de una aplicación web.....	29
3.2.2	Utilización de las tecnologías Web 2.0.....	30
3.2.3	Utilización de las aplicaciones de voz cómo recursos de aprendizaje.....	31
3.3	Recursos de aprendizaje	32
3.4	Herramientas tecnológicas para despliegue y desarrollo de la propuesta	33
3.4.1	Marcos de trabajo	34
3.4.2	Versiones de herramientas tecnológicas.....	36
3.5	Resumen del capítulo	37
Capítulo 4	Diseño de la propuesta.....	38
4.1	Arquitectura del sistema	38
4.2	Casos de uso	39
4.3	Diagrama de clases.....	41
4.4	Diagramas de robustez.....	42
4.4.1	Diagramas de robustez de examen oral.....	43

4.4.2	Diagramas de robustez de examen	44
4.4.3	Diagramas de robustez de blog	45
4.4.4	Diagramas de robustez de agregar anuncios.....	46
4.4.5	Diagramas de robustez de evaluación de exámenes.....	47
4.4.6	Diagramas de robustez de administración de aportaciones	48
4.5	Diseño de recursos de aprendizaje	48
4.5.1.1	Diseño de exámenes	49
4.5.1.2	Diagrama de secuencia de exámenes.....	52
4.5.2.1	Diseño de exámenes orales.....	53
4.5.2.2	Diagrama de secuencia de exámenes orales	55
4.5.3.1	Diseño de blog.....	55
4.5.3.2	Diagrama de secuencia de blog	57
4.5.4.1	Avisos del profesor.....	58
4.5.4.2	Diagrama de secuencia de avisos del profesor.....	60
4.6	Diseño de las funcionalidades haciendo uso de las tecnologías Web 2.0.....	60
4.6.1	AJAX.....	60
4.6.2	XSLT	62
4.6.3	JQUERY	64
4.7	Otras consideraciones de diseño, creación o administración.....	64
4.8	Resumen del capítulo	67
Capítulo 5	Vistas y operación de la aplicación.....	68
5.1	Vista de inicio de sesión	68
5.2	Vista de exámenes evaluados y ejercicios creados.....	69
5.3	Vista de creación de exámenes.....	71
5.4	Vista de creación de exámenes orales	73
5.5	Administración de aportaciones y de respuestas orales.....	74
5.6	Vista de creación de blog	74
5.7	Vista de anuncios	75
5.8	Forma de registro y lista de estudiantes	76
5.9	Resumen del capítulo	77
Capítulo 6	Conclusiones y trabajo a futuro.....	78

6.1	Introducción.....	78
6.2	Problemas encontrados	79
6.3	Experiencias aprendidas	80
6.4	Conclusiones.....	81
6.5	Trabajo a futuro.....	83
	Bibliografía.....	84
	Anexo Configuración y uso de AJAX.....	89
	Anexo Generador de Exámenes	91
	Anexo Generador General de Exámenes	93
	Anexo Generador de datos Exámenes orales, Blog y Anuncios.....	96
	Anexo Documento VXML consistente en Examen.....	101
	Anexo Plantilla XSL para creación de Examen oral.....	102
	Anexo Plantilla XSL para creación de Blog.....	103
	Anexo Documento XML con datos de Blog.....	105
	Anexo Documento XML con datos de Anuncios.....	105
	Anexo Documento XML con datos de Examen oral.....	105
	Anexo Documento VXML consistente en Examen oral	106
	Anexo Documento VXML consistente en Anuncios	107
	Anexo Servlet para evaluación de exámenes.....	108
	Anexo Servlet para salvado de respuestas orales.....	111
	Anexo Servlet para salvado de aportaciones	114
	Anexo Diagrama de flujo de evaluación de un examen	116
	VITA.....	117

Índice de Figuras

FIGURA 2. 1	MAPA CONCEPTUAL DE CONTENIDO DEL CAPÍTULO	8
FIGURA 2. 2	EJEMPLO DE APLICACIONES WEB 2.0	17
FIGURA 2. 3	TECNOLOGÍAS WEB 2.0 (ADAPTADO DE NOUREDDINE Y DAMODARAN, 2008)	17
FIGURA 2. 4	EJEMPLO MAJAX	19
FIGURA 2. 5	EJEMPLO DE TRANSFORMACIÓN XSLT ADAPTADO DE (BURKE, 2001)	20
FIGURA 2. 6	ELEMENTO XSL:TEMPLATE	21
FIGURA 2. 7	ELEMENTO XSL:TEMPLATE SIN PUNTO	21
FIGURA 2. 8	DIAGRAMA DE TELEFONIA Y RED DE DATOS CON SISTEMA IVR	25
FIGURA 2. 9	MALINTERPRETACIÓN DE PALABRAS.....	26
FIGURA 2. 10	MENÚ COMO CONTROL DE FLUJO DE LA LLAMADA.....	26
FIGURA 3. 1	MECANISMO DE PROCESAMIENTO DE PETICIONES DE SPRING MVC	35
FIGURA 3. 2	ARQUITECTURA DE ALTO NIVEL DE HIBERNATE.....	35
FIGURA 3. 3	DIAGRAMA DE COMUNICACIÓN DEL NAVEGADOR Y EL SERVIDOR WEB CON DWR	36
FIGURA 4. 1	ARQUITECTURA DEL SISTEMA	38
FIGURA 4. 2	DIAGRAMA DE CASOS DE USO	40
FIGURA 4. 3	DIAGRAMA DE CLASES DEL SISTEMA	41
FIGURA 4. 4	DIAGRAMA DE ROBUSTEZ DE CREACIÓN DE EXÁMENES ORALES.....	43
FIGURA 4. 5	DIAGRAMA DE ROBUSTEZ DE CREACIÓN DE EXÁMENES	44
FIGURA 4. 6	DIAGRAMA DE ROBUSTEZ DE CREACIÓN DE BLOG.....	45
FIGURA 4. 7	DIAGRAMA DE ROBUSTEZ DE AGREGAR ANUNCIOS	46
FIGURA 4. 8	DIAGRAMA DE ROBUSTEZ DE EVALUACIÓN DE EXÁMENES	47
FIGURA 4. 9	DIAGRAMA DE ROBUSTEZ DE ADMINISTRACIÓN DE APORTACIONES	48
FIGURA 4. 10	ELEMENTOS FORM, FIELD, GRAMAR Y FILLED (ADAPTADO DE VOXEO, 2011)	50
FIGURA 4. 11	EJEMPLO DE FORMA DEL LENGUAJE VXML COMO EXAMEN	51
FIGURA 4. 12	DIAGRAMA DE SECUENCIA DE EXÁMENES	52
FIGURA 4. 13	EJEMPLO DE FORMA DEL LENGUAJE VXML COMO EXAMEN ORAL	54
FIGURA 4. 14	DIAGRAMA DE SECUENCIA DE EXÁMENES ORALES	55
FIGURA 4. 15	EJEMPLO DE FORMA DEL LENGUAJE VXML COMO BLOG.....	56
FIGURA 4. 16	DIAGRAMA DE SECUENCIA DE BLOG	57
FIGURA 4. 17	EJEMPLO DE FORMA DEL LENGUAJE VXML PARA MUESTRA DE ANUNCIOS	58
FIGURA 4. 18	SCRIPT PARA ANÁLISIS DE DATOS DE ARCHIVO XML	59
FIGURA 4. 19	EJEMPLO DE ARCHIVO XML CON DATOS DE ANUNCIOS DE UN PROFESOR.....	59
FIGURA 4. 20	DIAGRAMA DE SECUENCIA DE AVISOS DEL PROFESOR	60
FIGURA 4. 21	DIAGRAMA DE SECUENCIA ACTUALIZACIÓN DE EJERCICIOS PRESENTADOS.....	61
FIGURA 4. 22	NAMESPACES Y ESQUEMAS EN EL SERVLET DEL MARCO DE TRABAJO SPRING	61
FIGURA 4. 23	REPRESENTACIÓN DE TRANSFORMACIÓN XSLT (ADAPTADO DE BURKE, 2001)	62
FIGURA 4. 24	TRANSFORMACIÓN DE DOCUMENTOS VXML USANDO LA TECNOLOGÍA XSLT.....	62
FIGURA 4. 25	REPRESENTACIÓN DE TRANSFORMACIÓN XSLT (ADAPTADO DE BURKE, 2001)	63
FIGURA 4. 26	CÓDIGO DE ACTUALIZACIÓN DE DATOS XML.....	63
FIGURA 4. 27	JAVASCRIPT Y JQUERY. CREACIÓN DINÁMICA DE PREGUNTAS	64
FIGURA 4. 28	EJEMPLO DE CÓDIGO PARA ASIGNACIÓN DE NOMBRES A ARCHIVOS DE AUDIO.....	65
FIGURA 4. 29	EL GENERADOR DE EXÁMENES REUTILIZA CÓDIGO DE UN GENERADOR MÁS GENERAL.....	66
FIGURA 4. 30	DOM4J PARA ADICIÓN DE NUEVOS NODOS DIRECTAMENTE AL DOCUMENTO XML.....	67

FIGURA 5. 1	VISTA DE INICIO DE SESIÓN	68
FIGURA 5. 2	VISTA DE REGISTRO DE PROFESORES	69
FIGURA 5. 3	VISTA DE EXÁMENES EVALUADOS Y EJERCICIOS CREADOS	69
FIGURA 5. 4	DESPLIEGUE DE MENÚ. ORGANIZACIÓN DE RECURSOS DE APRENDIZAJE.	70
FIGURA 5. 5	VISTA DE CREACIÓN DE EXÁMENES	71
FIGURA 5. 6	MENSAJE DE ALERTA DE LÍMITE MÁXIMO DE PREGUNTAS.	71
FIGURA 5. 7	EJEMPLO DE MENSAJE DE ALERTA DE VALIDACIÓN DE CAMPOS.	72
FIGURA 5. 8	EJEMPLO DE DOCUMENTO VXML (EXAMEN) CREADO POR LA APLICACIÓN.....	72
FIGURA 5. 9	VISTA DE CREACIÓN DE EXÁMENES ORALES.....	73
FIGURA 5. 10	EJEMPLO DE DOCUMENTO VXML (EXAMEN ORAL) CREADO POR LA APLICACIÓN.....	73
FIGURA 5. 11	REPRODUCCIÓN DE AUDIO. RESPUESTAS ORALES DADAS POR ALUMNOS.....	74
FIGURA 5. 12	VISTA DE CREACIÓN DE BLOG.....	75
FIGURA 5. 13	VISTA DE ANUNCIOS	75
FIGURA 5. 14	EJEMPLO DE DOCUMENTO XML DE DATOS DE ANUNCIOS	76
FIGURA 5. 16	FORMA DE REGISTRO DE ESTUDIANTES	76
FIGURA 5. 17	VISTA DE LISTA DE ESTUDIANTES	77

1.4 Objetivos de la investigación

Los cuatro objetivos de investigación en este trabajo son:

- Análisis de la relevancia de la propuesta como proyecto de aprendizaje móvil.
- Análisis y determinación de cómo pueden ser usadas las tecnologías computacionales y el lenguaje VXML en la construcción de una aplicación computacional.
- Desarrollo de un prototipo de aplicación computacional que permite la creación y administración de recursos de aprendizaje.
- Análisis de las ventajas y desventajas de la aplicación computacional como propuesta para la creación y administración de recursos de aprendizaje.

1.5 Preguntas de investigación

Las preguntas de investigación tienen la finalidad de guiar el contenido del marco teórico y obtener los conocimientos necesarios para generar una propuesta adecuada para el cumplimiento de los objetivos especificados. A continuación se presenta un listado de ellas:

- ¿Cómo se define el aprendizaje móvil?
- ¿Cuáles son las características del aprendizaje móvil?
- ¿Cuáles son los antecedentes de proyectos de aprendizaje móvil?
- ¿Qué se puede aprender a partir de los elementos de éxito o fracaso de los proyectos de aprendizaje móvil consultados?
- ¿Cuáles son las tecnologías Web 2.0?
- ¿Cuál es la relevancia de las tecnologías Web 2.0 sobre sus predecesoras?
- ¿Para qué sirve Ajax?
- ¿Para qué sirve la librería XSLT?
- ¿Para qué sirve la librería jQuery?
- ¿Cuál es la infraestructura necesaria para el desarrollo de una aplicación computacional?
- ¿Qué es el lenguaje VXML?
- ¿Cuál es la infraestructura necesaria para el despliegue de una aplicación de voz?
- ¿Qué consideraciones de diseño se deben tomar en cuenta en el desarrollo de aplicaciones de voz?

Estas preguntas serán resueltas a lo largo del contenido del marco teórico desarrollado en el siguiente capítulo.

2.1 Aprendizaje móvil

2.1.1 Definición

En la literatura se pueden encontrar distintas definiciones del aprendizaje móvil, dependiendo del enfoque dado. En uno de ellos se establece una asociación entre la tecnología móvil y el aprendizaje que se pueda obtener a partir de esta. Dicho enfoque es descrito en la siguiente definición:

“proceso de adquirir conocimiento y ser capaz de operar satisfactoriamente dentro y sobre, nuevos y siempre cambiantes contextos y espacios de aprendizaje” (Pachler, Bachmair, & Cook, 2010, pág. 6).

A partir de esta definición es importante reconocer que el individuo o usuario de la tecnología móvil no solo obtendrá conocimientos, sino que además, debe ser capaz de manejar la herramienta y estar preparado para cambios que se realicen tanto en el dispositivo, como en la forma de adquisición de los contenidos.

Otra definición que se le da al aprendizaje móvil, está más relacionado con el área o dominio al que pertenece. Esto es:

“el aprendizaje móvil es un campo educacional que surge a partir de la combinación o evolución del aprendizaje electrónico (e-learning) y el aprendizaje colaborativo soportado por computadora o CSCL por sus siglas en inglés. Por ser estas dos últimas aéreas, una evolución del aprendizaje a distancia, el aprendizaje móvil hereda la característica de transmisibilidad a distancia con la diferencia de que esta pueda cambiar de lugar, es decir de que es móvil” (Yordanova, 2007, pág. 1)

A partir de estas dos definiciones se puede abreviar una definición más completa la cual precisa al aprendizaje móvil, “como el aprendizaje a distancia que se obtiene mediante el uso de dispositivos móviles para la transmisión de conocimientos y de los cuales se conoce su operación o funcionamiento”.

2.1.2 Características

Las principales características del aprendizaje móvil son descritas por Pachler, et al (2010)

- Permite a los estudiantes el aprendizaje en cualquier lugar y en cualquier tiempo, permitiendo la recolección, el acceso y el procesamiento de la información fuera del aula de clase.

- Tienen la capacidad de llevar la educación a alumnos en comunidades en donde no es accesible implantar tecnologías que requerirían mayor infraestructura, esto gracias a la tecnología satelital.
- Promueven y fomentan la colaboración y comunicación.
- Habilitan experiencias de aprendizaje personalizadas. Es decir que la rapidez y la forma de aprendizaje se adecuan a las actividades y tiempos de los alumnos, ya que depende totalmente de las preferencias de estos, el elegir cuándo y qué recursos revisar.

A diferencia del *aprendizaje electrónico* (e-learning), el aprendizaje móvil agrega la característica de movilidad y de “aprendizaje personalizado” (Ally, 2010, pág. 14).

La característica de personalización, puede ser observada en la capacidad del dispositivo móvil de contener solo aquellos recursos de aprendizaje previamente seleccionados de acuerdo a las preferencias de estudio de los usuarios, aunque también debido a las limitaciones de la misma tecnología, por ejemplo contenido solo de audio debido a los todavía limitados tamaños de pantalla de los dispositivos móviles para la reproducción de texto, figuras y video (Koole, 2006, pág. 1).

2.1.3 Modelos e impacto en los alumnos

Si bien el objetivo principal y la filosofía del aprendizaje móvil, es la adquisición de conocimientos a cualquier hora y lugar, el uso de las tecnologías involucradas permite transformar la experiencia del estudiante (UNESCO, 2005, pág. 31). Un ejemplo de ello es la capacidad de reproducción sin límite de veces de recursos de aprendizaje en formato de video o audio.

Esta forma de aprender ha orillado a la comunidad científica a desarrollar modelos en donde se establezca el proceso de adquisición de conocimiento mediante la tecnología móvil. Algunos de los elementos de estudio de un modelo de aprendizaje móvil son el control de la interacción humano-tecnología, el contexto del sistema o tecnología computacional usado, el contexto de los actores involucrados y las formas de comunicación que se produce en la interacción con la tecnología (Sharples, Taylor, & Vavoula, 2005, págs. 7-8).

Entre los principales objetivos de los modelos de aprendizaje del aprendizaje móvil, se encuentran el de servir como guía en el desarrollo de futuros dispositivos móviles, el diseño de recursos de aprendizaje o el desarrollo de estrategias de enseñanza para la educación móvil. Ejemplo de ello es el Modelo para el Análisis

- < - A
8 \$
< \$ A

□

□ < > / 8
7 AB
D

□

\$
\$
F A

□

/ \$
\$
8 HA

□

□ \$ KA
1 8 \$ 3
6 0 \$
/ 0 8 \$ A \$
J 1/ B K A/
\$: \$

□

□

□ : \$
8 \$
\$ D

□

\$ D 8 \$ 1 \$ 8

\$ 1 D \$ 7KA \$ 8

, \$ E 1 < \$ A

(D D 8 D & D L 3 0 8 8 8 8 A

8 \$ D \$ J

... \$

6 < 6 \$ \$A \$ 5 3 6 \$ 48 \$ - / A

, \$ J D

\$ 8 8 D \$ D 1 1

1 \$ 3 & D # 7A

0
0

& 8
D
<2 & # I 5 LK /
&

-
-
-
-
-
-
-
-
-
-



&
<# > # 7 FX

0
0

& 8
G
4 11 A

-
-
-
-
-
-
-
-
-

B
 (1
 & 8 \$ P F /
 &



-
-

L 0 & < J \$ # 5 /

E

□

/ \$ & 8 D FA

□

□

0

□

() < \$ 8 () A G0 (1 11 # B \$) A # > # 7 H CH A \$ D () \$ (1) & G F H GA

□

□

1 0

□

0 \$ 5A

□

4 \$) G- 8 8 5A

□

4) G0 8 \$ 8 \$ # B A & 8 # 5A

□

B 9

\$ \$

\$

& & & / # B A KA

4 E

\$

(1

< \$ > 8 5 A () 8 8

0

:

8

\$ < G

F FA G

() < \$ > = A (1BJ < 1BJ B- PA # & - # 6 / 0 & = - B 0 6 J P B- PA

```
//Cualquier función que se ejecute a partir de un evento en el documento
function useMajax(){
    var url = "data.txt";
    //Se realiza una petición HTTP a través del método GET
    //Se especifica el documento a abrir
    majax_get(url,null);
}
function handler(){
    //Se sobre escribe la información en el elemento del documento
    var div = document.getElementById("targetDiv");
    div.innerHTML = MAJAX_RESPONSE;
}
//Se registra la función de la llamada de regreso
MAJAXCM_COMPLETEE.register(handler);
```

H ()

< G FA

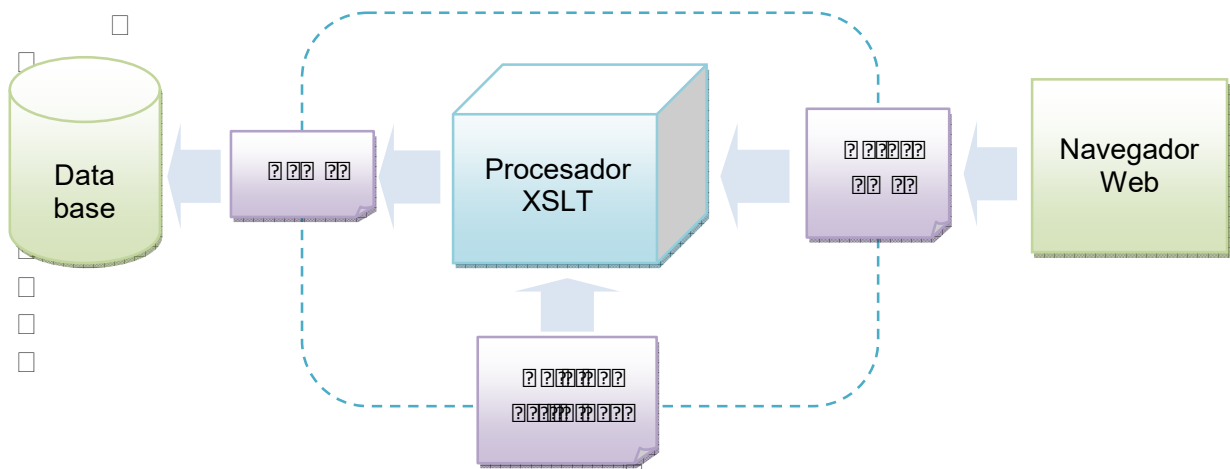
#& - (1 \$ 8) 1 0 < : 1 \$ 8 A 3) 0 4 # 12 P 2 -) 0 < A

23

) 1 0 < : 1 \$ 8) 1 0 < : 1 \$ 8 A 9 8) 1 C B & & & / 0 8 : 1 \$ 8 \$ 1 A A

4 1 23

4) 1 0 < : 1 \$ 8 A 9 8) 1 C B & & & / 0 8 : 1 \$ 8 \$ 1 A A



K) 1 0 < : 1 \$ 8 A

-
-
-
-
-

0 A

8) 1 \$) \$

) 1 0)

L # 8 : 9

```

<xsl:template match="/" >
  <xsl:apply-templates select="objeto"/>
</xsl:template>

<xsl:template match="objeto">
  <html><body><h1>
    <xsl: value-of select="." />
  </h1></body></html>
</xsl:template>

```

:

0 A

8 : 4 8 9 \$C

```

<xsl:template match="/" >
  <html><body><h1>
    <xsl:value-of
  select="objeto" />
  </h1></body></html>
</xsl:template>

```

F :

0 A

H 1 : 10 D 8 Q C R N N S

2.2.7 jQuery

jQuery es una librería construida sobre JavaScript que permite realizar tareas de scripting más complejas que si se codificaran en JavaScript (Castledine & Sharkie, 2010, p. 1). Además de que simplifica las tareas de codificación, otra de las razones por la que es importante el uso de jQuery, es para satisfacer las demandas actuales del rápido desarrollo de páginas Web, tales como la aceptación de entradas de usuario más complejas y su rápida respuesta, el desempeño de las mismas o requerimientos visuales de usuarios (Hall, 2009, pág. 67).

Algunas de las tareas que permite jQuery son descritas por Chaffer y Swedberg (2007), estas son:

- Acceso a los elementos del DOM
- Modificación de la apariencia de la página.
- Alterar el contenido de una página.
- Responder a interacciones del usuario dentro de la página.
- Agregar animaciones
- Regreso de información desde el servidor sin necesidad de refrescar la pagina.

Si bien algunas de las tareas pueden realizarse fácilmente a través de simple JavaScript, algunas de ellas, como por el ejemplo el desvanecimiento de un menú, serian difícilmente de realizarse sin jQuery. No se pueden realizar tareas con jQuery que no se logren con JavaScript, ya que son lo mismo (Castledine & Sharkie, 2010, pág. 8). JQuery solo simplifica las tareas.

2.2.8 Funcionamiento jQuery

La librería jQuery permite el acceso a una función JavaScript la cual tiene el mismo nombre de la librería y también contiene cientos de otras herramientas que permiten realizar las tareas de modificación de páginas. Puesto que las últimas son accedidas a través de la función “jQuery”, es que se utiliza el alias “\$”, con la intención de evitar una codificación más alargada debido a la palabra (Castledine & Sharkie, 2010, pág. 12).

La ejecución de una función usando jQuery se realiza siempre y cuando la página se ha cargado. No se puede manipular una página hasta que el documento esté listo, por lo que es necesario incluir el código `$(document).ready()` para poder detectar este estado (Murphey, 2010, pág. 22). Este primer pedazo de código es por lo tanto el primer elemento antes de escribir cualquier función usando la librería.

9
D TA
TUA
D
TVA
TA
D
8 > 1 F 5A

\$A \$
D T< A 8 8
8 \$
T
8 T <
D D TA < \$A\$
D < >
18 A

8 9 .

&
B 8
B 77H LFA
D & \$
G0

1
E 8
8 \$
8 < P # 3 >
4 5A

-
-
-
-
-
-
-
-
-
-
-

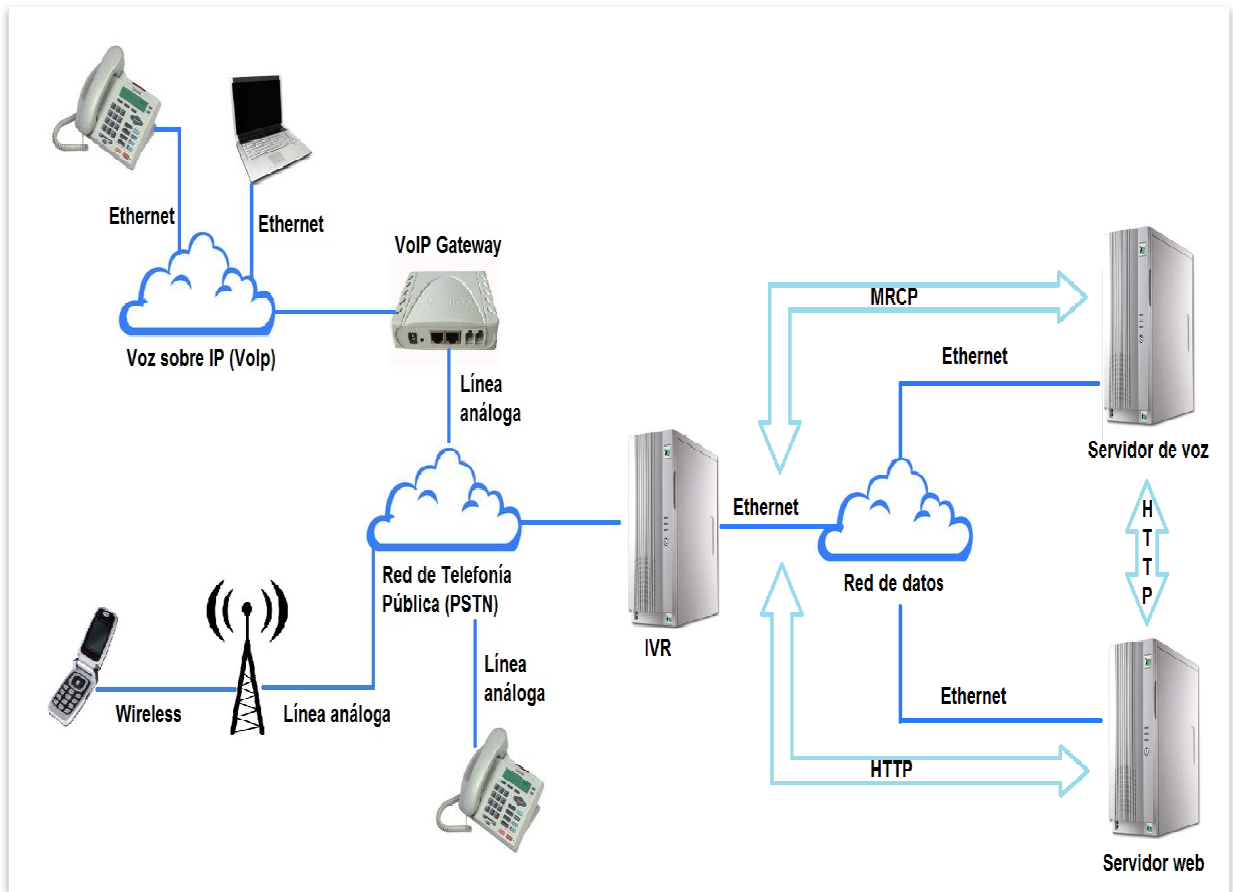
2

2) \$ & & / < & L / A G0 < A \$ D \$ D < K LA

5 \$ < 2 64A E < 10J A < 8 KA

62 - < 6 2 A \$ 62 - D - / 4 < A

B 2) \$ P 2) - 6 < 6 A 2) D < & A < 6 K HA



5 # 0 \$ # 62 -

< / 8 K A

1- A # <001A D D: < 8 K FA

3 2) \$ 8 \$ 8 : < > = \$ /

Entre los principales objetivos que se espera alcanzar con el diseño de las aplicaciones de voz, es el de presentar mejor la información para una experiencia placentera del usuario, manejar o predecir errores de pronunciación para evitar problemas de obtención de datos, reducir los costos de llamadas al hacer más eficiente la selección de opciones o construir una confiabilidad en el usuario al existir todos los elementos anteriores (IBM, 2005, págs. 9-13).

Las siguientes son algunas de las sugerencias de diseño que son presentadas por IBM en su Guía para Programadores de VXML (2005):

- Tomar en consideración la forma de cómo dirigirse al usuario, pudiendo ser con un hablado cordial u otro como por ejemplo uno más técnico.
- Tener en consideración los comandos de salida. Por ejemplo los usuario pueden mencionar la palabra salir pero no estar seguros de su decisión, para lo cual sería necesario implementar un dialogo de confirmación de salida.
- Selección de modo de interrupción. Para los casos de presentar información muy importante deshabilitar la opción de interrupción.
- Identificación de reconocimiento de problemas. Esto es detección temprana de posibles errores de pronunciación, como sucede en el siguiente ejemplo:

```
Sistema: Vuelo desde
Usuario: Los Ángeles <no confundible>
Sistema: Vuelo hacia
Usuario: Newark <marcado como confundible>
Sistema: Newark, New Jersey o New York, New York
Usuario: Newark, New Jersey
```

Figura 2. 9 Malinterpretación de palabras
(Adaptado de IBM, 2005)

- En caso de solicitud de información como en servicios de centro de llamadas, proporcionar menús que permitan un control de la conversación.

```
Sistema: Seleccione Teléfono o Fax
Usuario: Teléfono
Sistema: Seleccione acciones Telefónicas. Dejar Mensaje. Llamar
```

Figura 2. 10 Menú como control de flujo de la llamada
(Adaptado de IBM, 2005)

- Apropiaada interfaz de usuario. En caso de hacer uso de números en los diálogos utilizar diálogos de presión de teclado o DTMF – Dual Tone Multi-Frequency.

Para un primer caso, la aplicación web, representa la plataforma por la cual se pueden crear o elaborar los recursos de aprendizaje, también podría visualizarse por parte de los profesores como una herramienta de creación. Sin embargo, desde el punto de vista de los programadores, lo que contribuye a la creación de los recursos de aprendizaje son las tecnologías computacionales y las tecnologías Web 2.0.

En el caso de las aplicaciones de voz desarrolladas con el lenguaje VXML, estas representan los recursos que serán usados para la obtención de un aprendizaje a través del audio. Por ser estos accesibles a través de dispositivos móviles, es que se considera la propuesta como un proyecto de aprendizaje móvil.

Aunque la propuesta establece la creación de recursos de audio, esto no significa que se proponga o se de justificación a la sola utilización de este tipo de recursos de aprendizaje, en un curso de aprendizaje móvil. La propuesta se visualiza como complementaria para aquellos cursos en donde se presenten las limitantes tecnológicas de los teléfonos celulares con pocas capacidades de procesamiento o pequeño tamaño de pantalla para la reproducción de videos.

3.2.1 Utilización de una aplicación web

Las razones de utilización de una aplicación web como propuesta, se pueden visualizar en las características que se presentan en estos sistemas, algunas de las cuales son:

- Se encuentran disponibles a cualquier hora y en cualquier lugar disponiendo de una computadora con acceso a internet (Darie, Brinzarea, Chereches-Tosa, & Bucica, Building Responsive Web Applications AJAX and PHP, 2006, pág. 9).
- Contienen diversos métodos de entrada de datos (campos de texto, botones, botones de radio, casillas de listado, etc.).
- Proveen varios mecanismos de control de navegación como por ejemplo un menú de herramientas (Kumar, Agarwal, & Manwani, 2010).
- No requieren de una instalación o actualización especial, solo es necesario el uso de un navegador de internet los cuales vienen pre instalados en virtualmente todas las computadoras (Shklar & rosen, Web Application Architecture Principles, Protocols and Practices, 2003, pág. 202).

- Fáciles de crecer en contenido. Una vez que se sube al servidor web un material o actualización estas son visibles por todos los usuarios (Darie, Brinzarea, Chereches-Tosa, & Bucica, Building Responsive Web Applications AJAX and PHP, 2006, pág. 9).

Por lo anterior, los profesores tendrán a la mano un interfaz visual que les provea de diversas herramientas de control y de ingreso de datos para la creación de recursos de aprendizaje, además de que solo requerirán el conocimiento de navegación por internet y no necesitaran tener conocimiento tecnológico del lenguaje VXML.

3.2.2 Utilización de las tecnologías Web 2.0

Como se mencionó en el capítulo anterior, las tecnologías Web 2.0 permiten las construcción de aplicaciones web dinámicas, es decir aplicaciones que permiten una interacción entre el sistema y los usuarios, sin embargo, existen muchas otras razones del por qué utilizar las tecnologías.

Las tecnologías Web 2.0 son herramientas tecnologías de gran aceptación y rápida adopción por parte de las empresas en el mercado debido a las ventajas y características que ofrecen, tales como por ejemplo la facilitación del diseño Web, el re uso de aplicaciones y las actualizaciones, además porque permiten la creación de interfaces de usuario responsivas (Murugesan, 2007). En pocas palabras, facilitan el desarrollo y utilización de las aplicaciones web.

Algunos estudios demuestran que las aplicaciones desarrolladas con tecnologías Web 2.0 incrementan el entusiasmo de sus usuarios y sus habilidades tecnológicas (Begum & Gultekin, 2010). Dentro de este contexto, esto significaría que se generaría en los profesores una motivación para su uso.

No obstante, lo que se considera esencial del uso de las tecnologías, es su capacidad de permitir la creación de contenidos generados por usuarios y su capacidad de incrementar el compromiso de ellos, en la creación y manipulación de la información de los contenidos (Anderson, 2007, pág. 15).

El uso de las tecnologías permite también agregar la característica de mantenibilidad a la aplicación. Por ejemplo con XSLT, se puede conservar todos los datos actualizables en un solo documento XML. Lo anterior permite un mantenimiento más sencillo, debido a la separación entre los datos y las estructuras de presentación de ellos (Key, 2008, pág. 34). De este modo, cuando exista la necesidad de un cambio o arreglo en la estructura de presentación del contenido, solo será necesario hacer el cambio en un solo documento, la hoja de estilo.

DWR, también es otro marco de trabajo para la implementación de Ajax. La librería permite además de la interacción de funciones Java Script a funciones de Java, la interacción inversa de la mencionada, también conocida como Ajax en reversa (DirectWebRemoting).

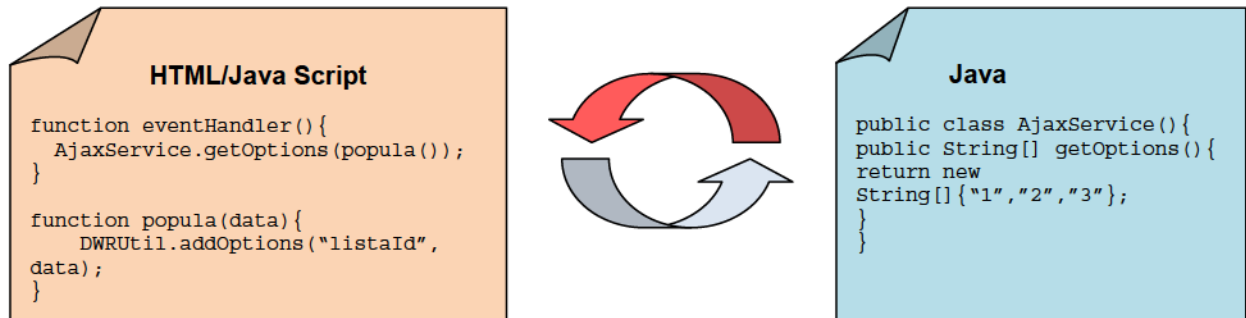


Figura 3. 3 Diagrama de comunicación del navegador y el servidor web con DWR (Adaptado de DirectWeb Remoting)

3.4.2 Versiones de herramientas tecnológicas

Las versiones de las herramientas tecnológicas para despliegue y desarrollo del prototipo de aplicación web seleccionadas se encuentran enlistadas a continuación:

- Java Development Kit 1.6.0_19
- Java Runtime Environment 6
- Apache Tomcat 6.0.18
- MySQL 5.1.51 con Conector Java
- Navicat 9.0.15
- Eclipse Ganymede 3.4.2

Las versiones de los marcos de trabajo escogidos para el desarrollo de la aplicación, las librerías de las tecnologías Web 2.0 establecidas en el capítulo de marco teórico y otras librerías para el correcto funcionamiento de los marcos de trabajo, se presentan en la siguiente lista:

- | | |
|--|--------------------------------|
| • spring.jar – versión 2.0 | • commons-fileupload-2.0.1.jar |
| • hibernate3.jar | • commons-io-1.2.2.jar |
| • dwr.jar – versión 2.0 | • dom4j-1.6.1.jar |
| • jquery-1.3.2.js | • jstl.jar |
| • sax2.jar (para transformaciones XSL) | • jta.jar |
| • commons-collections-3.2.1.jar | • servlet-api.jar |
| | • standard.jar |

- **Users.**- Esta clase es utilizada para guardar la información de los usuarios, que son los profesores que utilizarían el sistema para la creación de recursos de aprendizaje. Los atributos que mantiene la clase guardan relación con datos personales e información referente a su experiencia en la enseñanza (años de experiencia y experto en alguna área). Un profesor puede crear de ninguno a muchos ejercicios y anuncios.
- **Exercises.** Es la clase que mantiene la información del tipo de ejercicio que está siendo creado. El campo de tipo puede mantener los valores de examen, examen oral o blog, de los cuales solo los dos primeros cuentan con preguntas y pueden ser calificados.
- **Questions.** Contiene la información de los ejercicios que si requieren preguntas, aunque como los exámenes orales, pueden no tener respuesta.
- **Students.** Con esta clase los profesores pueden mantener la información de los estudiantes que utilizan las aplicaciones de voz para obtener o evaluar un aprendizaje. Los alumnos cuentan con un campo de calificación global (GPA por sus siglas en inglés), el cual está relacionado con cada calificación independiente que obtenga debido a la toma de algún ejercicio que pueda ser evaluado.
- **Grades.** Es la clase que mantiene la calificación independiente de cada ejercicio que el alumno haya realizado. Esta calificación es asignada directa por el sistema en caso de evaluación de tipo examen o asignada directamente por el profesor en caso de cualquier otro tipo de ejercicio.
- **StudentsAnswers.** A través de la clase se mantiene las respuestas de los alumnos. Solo los ejercicios de exámenes podrán guardar las respuestas dadas por los estudiantes, los otros dos tipos de ejercicios (examen oral y blog), guardaran la respuesta del alumno en formato de audio, para lo cual el profesor puede asignar o no una calificación.
- **Announcements.** Guarda la información de los mensajes y fechas de los avisos creados por el profesor.

4.4 Diagramas de robustez

Los diagramas de robustez son diagramas de UML utilizados para el análisis paso a paso de los casos de uso, con el propósito de darle validez a la lógica de negocio en ellos, esto es poderle dar robustez a los casos de uso (Amber, Robustness Diagrams, 2009).

Como se mencionó, los diagramas de robustez que se presentarán corresponden a los casos de uso de las funcionalidades relacionados con la creación y administración de recursos de aprendizaje. En los siguientes diagramas, los elementos de interfaz representan las páginas web para presentación u obtención de información, los elementos de control, las clases de procesamiento de la información

Diagrama de Secuencia de Interacción

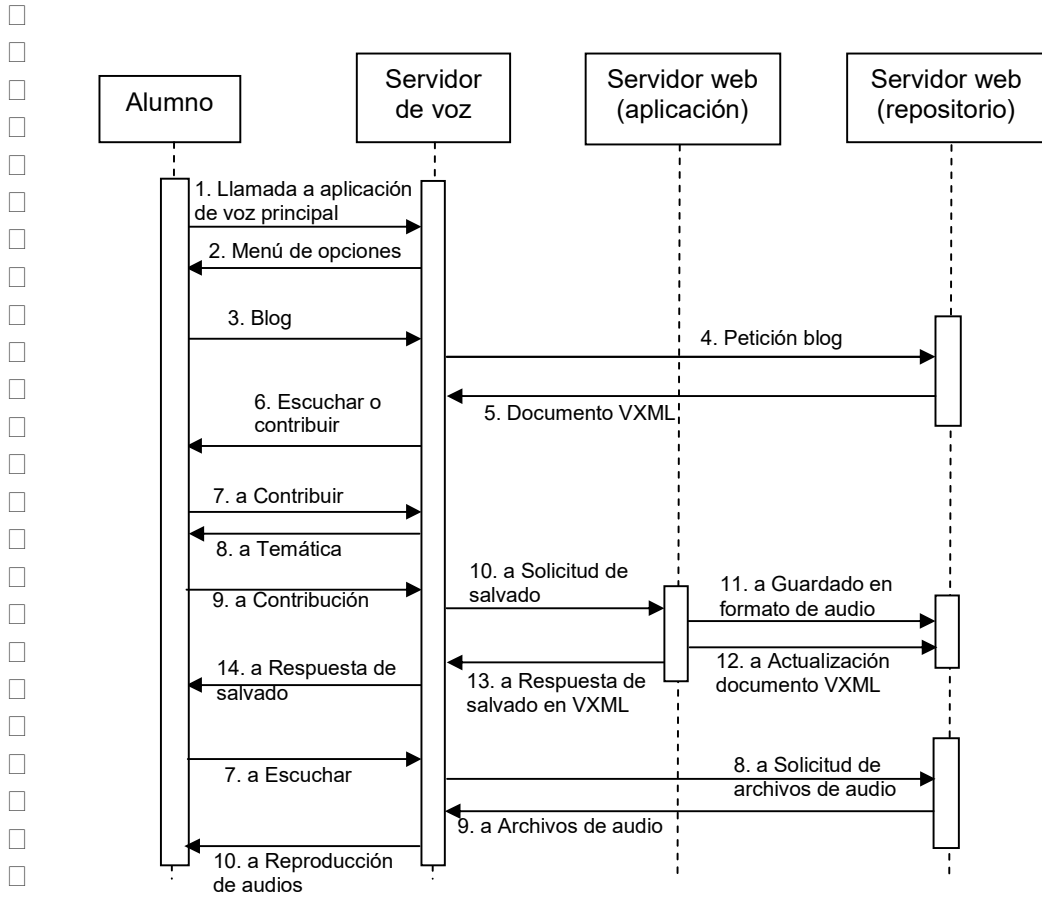


Diagrama de Secuencia de Interacción

Diagrama

Diagrama de Secuencia de Interacción

Diagrama

Diagrama de Secuencia de Interacción

Diagrama

interacción de los sistemas, el servidor de voz y la aplicación web, cuando cada recurso de aprendizaje es accedido. Y por último, se presentaron algunas consideraciones de diseño, administración y creación de los recursos de aprendizaje, que debían ser tomados en cuenta durante la etapa de implementación.

Las vistas de la aplicación fueron presentadas con el propósito de visualizar los elementos contenidos en las páginas de la aplicación y la estructura que se había seleccionado para la recolección de datos. Se hizo referencia a los procesos internos de la aplicación, cuyos códigos de programación pueden ser vistos en los anexos.

6.2 Problemas encontrados

Durante el desarrollo del prototipo de aplicación web se tuvieron varios problemas relacionados al uso de las tecnologías web. La solución a estos problemas se considera útil para aquellos desarrollos de aplicaciones que hagan uso de estas tecnologías y por lo tanto se presentan a continuación.

- A pesar de que con la librería commons-fileupload-1.2.2.jar, no existen problemas de compilación en la codificación de extracción de archivos de audio del objeto de petición (request en inglés), cuando es utilizado el servlet de examen oral o de blog, si existe un error cuando es ejecutada esta parte de código. El error se debe a que es necesaria una clase referenciada dentro de la librería commons-fileupload-1.2.2.jar y que está contenida en commons-io-2.0.1.jar, por lo que es necesario contar con la última.
- Un problema que se encontró en el envío de información referente a exámenes orales o blog, del servidor de voz a la aplicación web, es que no se puede enviar datos que no sean de contenido multiparte (ejemplo de contenido multiparte son los archivos de audio). Por lo tanto no se podrá obtener datos provenientes de los parámetros indicados en la lista de parámetros del elemento submit, al menos no en la versión 2 del lenguaje VXML. En el contexto de este trabajo, es necesario que además de los archivos de audio se incluya información referente por ejemplo al identificador del alumno. La solución a este problema, es escribir los parámetros y sus valores, directamente en la liga del elemento submit, es decir la liga al servlet de la aplicación web que espera obtener los archivos de audio. Para lo anterior es necesario solicitar al alumno su identificador en una sesión de voz independiente y que la aplicación de voz envíe esta información a la aplicación web, para que regrese un documento VXML consistente en el examen oral o blog, con la liga actualizada con el identificador del alumno.

- Los campos de formas creados dinámicamente en una página web, no son reconocidos por Spring, cuando se utiliza el modelo MVC y es utilizado un controlador que extiende de la clase SimpleFormController del marco de trabajo. Una solución a esta incapacidad de la versión 2.0 de Spring, es la de utilizar como objeto de la forma de la página, una lista del tipo LazyList, que represente los campos que sean creados dinámicamente. La especificación de enlace de un objeto Java con la forma de la página web, se realiza en el documento del contexto de la aplicación del marco de trabajo. De esta manera la página web, gracias al marco de trabajo, podrá tomar en cuenta los nuevos campos creados dinámicamente e insertar los valores del usuario a cada elemento de la lista, sin necesidad de hacerlo manualmente.

6.3 Experiencias aprendidas

Debido al uso de la tecnología VXML y al desarrollo del prototipo de aplicación web, es que se adquirieron las siguientes experiencias:

- Siempre es necesario realizar una revisión de compatibilidad entre las versiones de las tecnologías web que se utilicen, ya que en una nueva versión de una tecnología, pueden dejarse de utilizar clases que son necesarias en otra.
- Cuando es requerido utilizar en un JSP, contenido web ubicado en un directorio fuera de WEB-INF, como por ejemplo una librería JavaScript o una imagen, es necesario registrar un mapeo de default en el archivo de despliegue y configuración de la aplicación (web.xml) y utilizarlo para acceder al recurso. Lo anterior es consecuencia del marco de trabajo de esconder o no permitir el acceso a contenido web fuera de WEB-INF para protección de datos.
- Las consideraciones de diseño de los documentos VXML son de gran importancia puesto que definen la estrategia de creación de las aplicaciones de voz en base a los elementos de la tecnología. Por lo anterior, se pueden tener aplicaciones de voz que funcionan de manera muy similar, sin embargo dependiendo de los elementos elegidos del lenguaje, se puede tener aplicaciones de voz con diferente grado de mantenibilidad.
- Las consideraciones de diseño de los contenidos de los recursos de aprendizaje, también son de gran importancia, puesto que de no ser tomada en cuenta la pronunciación del contenido dada por el servidor de voz, se puede malinterpretar el mensaje que se quiere dar. Ejemplo de estas consideraciones es la de separar las letras de aquellas palabras que representen siglas.

- Cuando se trabaja con funcionalidades de escritura de archivos, tal como lo es el salvado de los archivos de audio provenientes de aportaciones o respuestas de exámenes orales, es necesario que el servlet que implementa el servicio, además de estar dado de alta en el archivo del contexto de la aplicación, extienda de clases del marco de trabajo con acceso a dicho contexto. Esto es de vital importancia para poder conocer la ubicación de los archivos guardados.
- Los archivos de audio y los recursos de aprendizaje son de tamaño mínimo, no requieren mucho espacio y además pueden ser reutilizados.

6.4 Conclusiones

La principal aportación de este trabajo fue diseñar e implementar un prototipo de herramienta tecnológica que facilita la creación de documentos VXML, al mismo tiempo de proponer los diseños, estructuras o estrategias para que una vez desplegados en el servidor de voz, representen recursos de aprendizaje de audio, accesibles por teléfonos celulares.

A partir de lo implementado y de lo investigado sobre la tecnología VXML, se puede concluir que las aplicaciones de voz consistentes en recursos de aprendizaje, tienen un gran potencial para poder ser utilizadas en cursos de enseñanza que empleen el aprendizaje móvil. Lo anterior debido a que las aplicaciones de voz permiten un aprovechamiento adecuado de la capacidad auditiva de los teléfonos celulares, cuando son utilizados como medios de transmisión de información y a que la creación y administración de las mismas se puede llevar a cabo de manera sencilla, gracias a las tecnologías computacionales.

No obstante, el uso de la tecnología VXML en el área educativa debe darse de manera cuidadosa, ya que existen varios factores que deben tomarse en cuenta antes de su implementación y por los cuales no se afirma que el uso de aplicaciones de voz sea la única opción que deba tomarse en cuenta para proyectos de aprendizaje móvil. Uno de estos factores son los diferentes estilos de aprendizaje de cada persona. Los recursos de aprendizaje de audio no serían los más apropiados en personas con estilos de aprendizaje distinto al auditivo. Además también, de que no se estarían aprovechando las capacidades visuales de los teléfonos celulares o de los Smartphones, lo que significaría un desaprovechamiento tecnológico de estos dispositivos. Otro factor es la mayor atención o cuidado que debe ponerse en el diseño de los recursos de aprendizaje, ya que, como se señaló en el capítulo 2, un mal diseño puede provocar que el mecanismo de texto a voz no pronuncie de manera correcta lo que se quiere expresar o que la aplicación de voz falle por no tomar en cuenta en sus gramáticas palabras que pudieran ser pronunciadas por los alumnos.

6.5 Trabajo a futuro

Debido a que la aplicación web es un prototipo, desarrollado con la intención de demostrar el cómo se puede facilitar la creación de recursos de aprendizaje de audio, es que se presentan varias consideraciones de optimización y mejora para trabajos a futuro.

- Seleccionar diseños de vistas de la aplicación que permitan una mejor estructuración de formas y que permitan una mejor presentación de datos.
- Permitir operaciones de edición de los recursos de aprendizaje que ya hayan sido creados. Esto con el objetivo de permitir la corrección de errores no previstos en el contenido del recurso de aprendizaje.
- Permitir operaciones de selección y subida al servidor de archivos de audio, los cuales puedan ser escuchados en las aplicaciones de voz consistentes en recursos de aprendizaje.
- Agregar a la aplicación una funcionalidad de evaluación de exámenes orales (opción de menú, forma, tabla de archivos de audio disponibles, etc.).
- Agregar a la aplicación funcionalidades de creación de distintos tipos de recursos de aprendizaje no previamente considerados. Por ejemplo creación de aplicación de voz que haga uso del mecanismo de reconocimiento de voz para detectar la correcta pronunciación de las palabras durante el aprendizaje de un idioma.
- Diseñar plantillas de los documentos VXML con elementos nuevos o mejorados de la versión 3 del lenguaje VXML, solo si permiten a la aplicación de voz ser más eficiente en la reproducción de sonido.
- Embeber las funcionalidades de la aplicación web de este trabajo en otras plataformas open source de administración o creación, como por ejemplo Moodle, para permitir la creación de distintos tipos de recursos de aprendizaje de acuerdo a las características del contenido educativo o para permitir a los alumnos que dispongan de una mayor variedad de recursos y adecuarse a su estilo de aprendizaje.
- Agregar en la aplicación de voz principal o en alguna de las aplicaciones de voz consistentes en recursos de aprendizaje, elementos del lenguaje VXML, que funcionen como mecanismos de manejo de errores cuando existan problemas de conectividad con el servidor web, con el objetivo de hacer más robustas las aplicaciones de voz y de esa forma evitar posibles fallas o problemas en la transacción de la información que se esté recolectando.
- Realizar pruebas de efectividad en la obtención de un aprendizaje que tendría la utilización de este tipo de recursos, tomando en cuenta consideraciones como el nivel de educación, el nivel socioeconómico, la cultura tecnológica o el grado de aceptación de los alumnos.

Bibliografía

Ally, M. (2010). *Mobile Learning Transforming the Delivery of Education and Training*. Edmonton, AB: AU Press, Athabasca University.

Amber, S. (2009). *Robustness Diagrams*. Retrieved Marzo 2011, from Agile Modeling: <http://www.agilemodeling.com/artifacts/robustnessDiagram.htm>

Anderson, P. (2007). *What is Web 2.0? Ideas, technologies and implications for education*. JISC Technology and Standards Watch.

Apache Tomcat Heritage. (2011). Retrieved Febrero 2011, from [tomcat.apache.org](http://tomcat.apache.org/heritage.html): <http://tomcat.apache.org/heritage.html>

Bailey, A., & Back, G. (2010). *MAJAX A JavaScript library for AJAX access to a III Millennium Catalog*. Retrieved Febrero 2011, from [libx.org](http://libx.org/majax/index.html): <http://libx.org/majax/index.html>

Begum, A., & Gultekin, O. (2010). Advantage of using web 2.0 applications in class. *ITiCSE '10 Proceedings of the fifteenth annual conference on Innovation and technology in computer science education* (p. 308). New York, NY, USA : ACM.

Borodin, Y. (2006). A Flexible VXML Interpreter for Non-Visual Web Access. *ASSETS'06* (pp. 301-302). Portland, Oregon, USA: ACM.

Brusilovsky, P., & Miller, P. (1999). Web-based testing for distance education. *Proceedings of WebNet'99, World Conference of the WWW and Internet* (pp. 149-154). Honolulu, HI, USA: AACE.

Burke, E. (2001). *Java and XSLT*. O'Reilly.

Caines, A., & Buttery, P. (2010). You talking to me?: a predictive model for zero auxiliary constructions. *NLPLING '10 Proceedings of the 2010 Workshop on NLP and Linguistics: Finding the Common Ground* (pp. 43-51). Stroudsburg, PA, USA: Association for Computational Linguistics.

Castledine, E., & Sharkie, C. (2010). *Jquery Novice to Ninja*. USA: Sitepoint.

Chaffer, J., & Swedberg, K. (2007). *Learning jQuery*. USA: Pack Publishing Ltd.

Chamberlain, J., Kempny, G., Elliott, G., Manoy, S., Takenaka, R., Derksen, E., et al. (2005). *IBM WebSphere Voice Server for Multiplatforms V5.1.1/V5.1.2 Handbook*. IBM Corp.

Conole, G., & Culver, J. (2009). Cloudworks: Social networking for learning design. *Australasian Journal of Educational Technology* , 763-782.

Cote, S. W. (2010). *Hemi - JavaScript Framework*. Recuperado el Febrero de 2011, de Whitefrost: <http://whitefrost.com/Hemi/>

Cuadrado, F., Garcia, B., Dueñas, J., & Parada, H. (2008). Case Study on Software Evolution towards Service-Oriented Architecture. *22nd International Conference on Advanced Information Networking and Applications - Workshops* (págs. 1399-1404). Madrid, España: IEEE Computer Society.

Cummings, J., Massey, A., & Ramesh, V. (2009). Web 2.0 proclivity: understanding how personal use influences organization adoption. *Proceedings of the 27th ACM International Conference on Design of*

communication. Ney York, NY, USA: ACM.

D'Aniello, A., Masone, A., & Tammara, A. (2006). Technological Innovation within EDS Italia Software: Experience Report. *Proceedings of the Conference on Software Maintenance and Reengineering (CSMR'06)*. Naples, Italy: IEEE.

Darie, C., Brinzarea, B., Chereches-Tosa, F., & Bucica, M. (2006). *Building Responsive Web Applications AJAX and PHP*. Reino Unido: Packt Publishing.

Deitel, P., & Deitel, H. (2009). *Ajax, Rich Internet Applications y desarrollo Web para programadores*. Madrid, España: Pearson Education, Inc. publishing as Prentice Hall.

DirectWebRemoting. (n.d.). *Overview od DWR*. Retrieved Marzo 2011, from [directwebremoting.org: http://directwebremoting.org/dwr/introduction/index.html](http://directwebremoting.org/dwr/introduction/index.html)

DIRECTWEBREMOTING.ORG. (n.d.). *DWR Easy Ajax for Java*. Retrieved Febrero 2011, from [directwebremoting.org: http://directwebremoting.org/dwr/index.html](http://directwebremoting.org/dwr/index.html)

Draganova, c. (2009). Use of Mobile Phone Technologies in Learning. *ITiCSE '09 Proceedings of the 14th annual ACM SIGCSE conference on Innovation and technology in computer science education* (p. 399). New York, NY, USA: ACM.

Eberman, B., Carter, J., Meyer, D., & Goddeau, D. (2002). Building voiceXML browsers with openVXI. *WWW '02 Proceedings of the 11th international conference on World Wide Web* (págs. 713-717). New York, NY, USA: ACM.

Eclipse. (2011). *Enterprise Java*. Retrieved Marzo 2010, from [eclipse.org: http://www.eclipse.org/home/categories/index.php?category=enterprise](http://www.eclipse.org/home/categories/index.php?category=enterprise)

Eichten, M., & Ferrans, J. (2007). *The 2006 VoiceXML Forum Survey*. Retrieved Febrero 2011, from VoiceXML Forum: http://www.voicexml.org/sites/default/files/Review/May2007/features/VoiceXML_survey.html#top

Farley, J., Crawford, W., Malani, P., Norman, J., & Gethland, J. (2006). *Java Enterprise in a Nutshell*. Sebastopol, CA: O'Reilly Media, Inc.

Gibson, B. (2007). Enabling an accessible web 2.0. *Proceedings of the 2007 international cross-disciplinary conference on Web accessibility* (pp. 1-6). New York, NY, USA: ACM.

Groth, D., & Jeffrey, M.-M. (Febrero de 2010). Why an informatics degree? *Magazine Communications of the ACM*. New York, NY, USA: ACM.

Hall, C. A. (2009). Web presentation layer bootstrapping for accessibility and performance. *W4A '09 Proceedings of the 2009 International Cross-Disciplinary Conference on Web Accessibililty* (pp. 67-74). New York, NY, USA: ACM.

Harms, D. (2007). A brief comparison of technologies available in the computer science classroom. *CompSysTech '07 Proceedings of the 2007 international conference on Computer systems and technologies* (pp. IV. 14-1 - 14-6). New york, NY, USA: ACM.

Helic, D., Maurer, H., Lennon, J., & Scerbakov, N. (2001). Aspects of a Modern WBT System. *In Proc. Intl. Conf. on Advances in Infrastructure for Electronic Business, education, Science, and Medicine on the Internet, SSGRR*.

Hibernate. (n.d.). *hibernate*. Retrieved Marzo 2011, from hibernate.org: <http://www.hibernate.org/about>

Holzner, S. (2007). *Ajax Bible*. Indianapolis, Indiana, USA: Wiley.

IBM. (2005). *Voice XML Programmer's Guide*. International Business Machines Corporation.

Jaschke, L. (2010, Septiembre). *Let's Get Informal: Mobile Phones for Adult Basic Education in West Africa*. Retrieved 2010, from Educational Technology Debate: <http://edutechdebate.org/meducation-initiatives/lets-get-informal-mobile-phones-for-adult-basic-education-in-west-africa/>

Johnson, R., Hoeller, J., Donald, K., Sampaleanu, C., Harrop, R., Arendsen, A., et al. (2011). *Spring Framework Reference Documentation*. Retrieved Marzo 2011, from springsource.org: <http://static.springsource.org/spring/docs/3.1.0.M1/spring-framework-reference/html/>

JSON.ORG. (n.d.). *Introducing JSON*. Retrieved Febrero 2011, from Json.org: <http://www.json.org/>

Key, M. (2008). *XSLT 2.0 and CPath 2.0 Programmer's Reference*. Indianapolis, Indiana, USA: Wiley Publishing, Inc.

King, G., Bauer, C., Andersen, M., Bernard, E., Ebersole, S., & Ferentschik, H. (2004). *Hibernate Reference Documentation*. Recuperado el Marzo de 2011, de jboos.org: http://docs.jboss.org/hibernate/core/3.6/reference/en-US/html_single/#tutorial-firstapp

Khairullah, E., & Al-Aama, A. (2009). Improving Accuracy of Context Aware Data Replication In Mobile Computing Application. *Proceedings of MoMM2009* (págs. 498-503). Kuala Lumpur, Malaysia: ACM.

Kidd, T. (2010). *Online Education and Adult Learning: New frontiers for Teching Practices*. Hershey, PA: IGI Global.

Kolias, C., Vassilis, K., Anagnostopoulos, L., Kambourakis, G., & Kayafas, E. (2008). A pervasive Wiki application based on VoiceXML. *Proceeding PETRA '08 Proceedings of the 1st international conference on PErvasive Technologies Related to Assistive Environments*. New York, NY, USA: ACM.

Koole, M. L. (2006). Practical Issues in Mobile Education. *WMTE '06 Proceedings of the Fourth IEEE International Workshop on Wireless, Mobile and Ubiquitous Technology in Education*. Washington, DC, USA: IEEE Computer.

Kristaly, D., Sisak, F., Truican, I., Moraru, S.-A., & Sandu, F. (2008). Web 2.0 technologies in web application development. *Proceeding of 1st international conference on Pervasive Technologies Related to Assistive Environments*. New York, NY, USA: ACM.

Kumar, A., Agarwal, S., & Manwani, P. (2010). *The Spoken Web Application Framework User Generated Content and Service Creation through lowend*. Raleigh, USA: ACM.

Kumar, A., Agarwal, S., & Manwani, P. (2010). The Spoken Web Application Framework. *19th International World Wide Web Conference*. Raleigh, USA: ACM.

Liu, C.-H., & Chen, W.-Y. (2009). The effectiveness of the handheld devices based on mobiles learning. *ICIS '09 Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human* (págs. 400-405). New York, NY, USA: ACM.

Lucas, B. (2000, Septiembre). VOICEXML for Web-Based Distributed conversational Applications. *Communications of the ACM Volume 43 Issue 9*. New York, NY, USA: ACM.

mber, S. (2010). *Introduction to UML 2 Class Diagrams*. Retrieved Marzo 2011, from agilemodeling.com: <http://www.agilemodeling.com/artifacts/classDiagram.htm#CompositionAssociations>

McLoughlin, C., & Reid, N. (2002). Seachange: Design of Online Quiz Questions to Foster Deep Learning. *Winds of Change in the Sea of Learning: Proceedings of the 19th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education* (pp. 843-846). Auckland: Williamson, C. Gunn, A. Young & T. Clear (Eds).

Murphey, R. (2010). *jQuery Fundamentals*. USA: Rebecca Murphey.

Murugesan, S. (2007). Understanding Web 2.0. *IT Professional Volume 9 Issue 4* , 34-41.

Navicat.com. (2011). Retrieved Marzo 2011, from http://www.navicat.com/en/products/navicat_mysql/mysql_overview.html

Oliver, M. S. (1994). *Code-Generation On-the-Fly: A Key to Portable Software*. Zurich, Switzerland: Ph.D. thesis, Swiss Federal Institute of Technology, Dissertation ETH No. 10497.

Oracle. (2010). *MySQL Enterprise Edition Database*. Retrieved 2011, from MySql.com: <http://www.mysql.com/why-mysql/white-papers/>

Oracle. (2010). *The Java EE Tutorial*. Oracle.

Pachler, N., Bachmair, B., & Cook, J. (2010). *Mobile Learning. Structures, Agency, Practice*. New York, NY, USA: Springer Science Business Media.

Reese, G. (1997). *Database Programming with JDBC and Java series*. Sebastopol, Calif.: O'Reilly.

Saade, R., & Bahl, B. (2004). The impact of cognitive absorption on perceived usefulness and perceived ease of use in on-line learning: an extension of the technology acceptance model. *Information & Management 42* (pp. 317-327). Canada: Elsevier B.V.

Sharples, M., Taylor, J., & Vavoula, G. (Septiembre de 2005). Towards a Theory of Mobile Learning. UK: University of Birmingham, The Open University.

Shklar, L., & rosen, R. (2003). *Web Application Architecture Principles, Protocols and Practices*. Chichester, England: John Wiley & Sons Ltd.

Skonnard, A., & Gudgin, M. (2003). *Essential XML Quick Reference*. Boston, MA, USA: Addison-Wesley.

SpringSource. (2011). *The Standard for Enterprise Java Development*. Retrieved Marzo 2011, from springsource.com: <http://www.springsource.com/developer/spring>

Stamey, J., & Richardson, T. (2006). Middleware development with AJAX. *Journal of Computing Sciences in Colleges Volume 22 Issue 2* , 281-287.

Sutter, J. (2009). *The top 10 tech trends 2009*. Retrieved Abril 2010, from CNN: <http://edition.cnn.com/2009/TECH/12/22/top.tech.trends.2009/index.html>

Tidwell, D. (2001). *XSLT*. Sevastapol, CA. USA: O'Reilly Media, Inc.

Tomei, L. (2008). *Adapting Information and Communication Technologies for Effective Education*. Hershey, PA, USA: Information Science Reference.

Trucano, M. (2010, Abril 4). *Mobile Phones and Literacy in Rural Communities*. Retrieved Noviembre 2010, from EduTech a World Bank Blog on ICT use in Education: <http://blogs.worldbank.org/edutech/mobile-phones-literacy>

Trucano, M. (2010, Septiembre). *Learning British English for the cost of a cup of Bangladesh tea*. Retrieved Noviembre 2010, from Educational Technology Debate: <http://edutechdebate.org/meducation-initiatives/learning-british-english-for-the-cost-of-a-cup-of-bangladesh-tea/>

UNESCO. (2005). Mobile Learning for Expanding Educational Opportunities. *International Workshop on Mobile Learning for Expanding Educational Opportunities* (p. 174). Tokyo, Japan: Asian Development Bank Institute.

Van Wamelen, J., & De Kool, D. (2008). Web 2.0: A Basis for the Second society? *Proceedings of the 2nd International Conference on Theory and Practice of Electronic Governance* (pp. 349-354). New York, NY, USA: ACM.

Voxeo. (2011). *VoiceXML Tutorials - Elements*. Recuperado el Febrero de 2011, de [vxml.org](http://www.vxml.org/):

w3.org. (2004). *Voice Extensible Markup Language (VoiceXML) Version 2.0*. Retrieved Marzo 2011, from w3.org: <http://www.w3.org/TR/voicexml20/>

World Wide Web Consortium, W. (2005). *Document Object Model (DOM)*. Obtenido de w3.org: <http://www.w3.org/DOM/>

World Wide Web Consortium, W. (2010). *The Extensible Stylesheet Language Family (XSL)*. Retrieved Febrero 2011, from w3.org: <http://www.w3.org/Style/XSL/>

Wright, J., & Dietrich, J. (2008). Survey of Existing Languages to Model Interactive web applications. *Conferences in Research and Practice in Information Technology Series; Vol. 325. Proceedings of the fifth on Asia-Pacific conference on conceptual modelling - Volume 79* (pp. 113-123). Darlinghurst, Australia, Australia: Australian Computer Society, Inc.

Yordanova, K. (2007). Mobile Learning and Integration of Advanced. *International Conference on Computer Systems and Technologies*. New York, NY, USA: ACM.

Yuschik, M. (2009, Junio). *A View from the Voice Search Conference*. Retrieved Febrero 2011, from Speech Technology: <http://www.speechtechmag.com/Articles/Column/The-View-from-AVIOS/A-View-from-the-Voice-Search-Conference-54011.aspx>

Ziya, A., & Ibrahim, H. (2004). An Audio-Book Project for Blind Students at the Open Education System of Anadolu University. *The Turkish Online Journal of Educational Technology – TOJET*, 3-10.

Anexo Configuración y uso de AJAX

Para la correcta operación de AJAX es necesario registrar las clases de servicio del lado del servidor en el archivo de contexto de la aplicación. Es necesario que se especifique si dentro del servicio se hará uso de una clase con acceso a la base de datos, la cual debe también estar registrada en el mismo archivo. También es necesario especificar cuál será el método de regreso en el elemento “include” y especificar donde se encuentra ubicada la clase del objeto de regreso en el elemento “convert”. La configuración en la aplicación web de este trabajo se muestra a continuación:

```
<dwr:configuration />
<dwr:controller id="dwrController" debug="true" />

<bean id="ExerciseService"
class="vxmlgenerator.service.ExerciseService">
<property name="gradeDao" ref="GradeDao" />
<property name="exerciseDao" ref="ExerciseDao" />
<dwr:remote javascript="ExerciseService">
    <dwr:include method="getExercisesTaken" />
    <dwr:convert type="bean" class="vxmlgenerator.domain.*">
    </dwr:convert>
</dwr:remote>
</bean>
```

El servicio AJAX de regreso de exámenes contestados es utilizado en el JSP de “home”. Para hacer uso del servicio es necesario especificar en código JavaScript que se utilizará un archivo con extensión “js” con el mismo nombre de la clase del servicio, este archivo no existirá, el mecanismo de AJAX se encargará de hacer la conversión del archivo para ser usado como código JavaScript. Será necesario especificar otros archivos de la librería ubicados en un directorio que no existe, aunque si será necesario crear un directorio con nombre “dwr” en el mismo nivel que el directorio WEB-INF. En este trabajo se utilizo la operación AJAX en un archivo JavaScript distinto. El código JavaScript de la página web de “home” se presenta a continuación:

```
<script type='text/javascript'
src='/vxmlgenerator/dwr/interface/ExerciseService.js'></script>
<script type='text/javascript' src='/vxmlgenerator/dwr/engine.js'></script>
<script type='text/javascript' src='/vxmlgenerator/dwr/util.js'> </script>
<script type="text/javascript" src='static/js/dwrExerciseServiceJS.js'>
</script>
<script type="text/javascript">
function timeRefreshed(period) {
    submitExercisesRequest();
    setInterval('submitExercisesRequest()', period);
}
</script>
</head>
```

```
<body onload="timeRefreshed(100000)">
```

El siguiente código es el archivo JavaScript del manejo de datos, en este archivo se realiza la llamada al método del servicio tal como si fuese código JavaScript, especificando como ultimo parámetro del método, el nombre del método de regreso, este parámetro no se especifica en el método de la clase del servicio.

```
function submitExercisesRequest() {
    var idX    =dwr.util.getValue("userId");
    var typeX  =dwr.util.getValue("type");

    var parametersX={userId: idX, exerciseType: typeX};

    //calling service, first objects are the parameter list
    //of the service method last parameter is the return
    //calling back method
    ExerciseService.getExercisesTaken(parametersX, showExercises);
}

function showExercises(responseX) {

    if (responseX.errorCode!=0){
        alert('Error: '+responseX.errorDescription);
        return;
    };

    var exercisesX=responseX.exercisesTaken;
    var lengthX=exercisesX.length;
    var temExerciseX;

    if (lengthX==0){
        return;
    };

    //delete rows except pattern row
    dwr.util.removeAllRows("myTable", {
    filter:function(tr) { return (tr.id != "myPattern");}});
    //dwr.util.setValue("serviceResponse", responseX.errorDescription);

    $("myHeaderPattern").style.display = "";

    var id="00";
    for (var i=0 ; i<lengthX ; i++){
        temExerciseX= exercisesX[i];
        id="00"+i;
        dwr.util.cloneNode("myPattern", { idSuffix:id });
        dwr.util.setValue("studentIdPattern" + id,
        temExerciseX.studentId);
        dwr.util.setValue("topicPattern" + id, temExerciseX.topic);
        dwr.util.setValue("timePattern" + id, temExerciseX.date);
        dwr.util.setValue("gradePattern" + id, temExerciseX.grade);
        $("myPattern" + id).style.display = "";
    }
}
```

Anexo Generador de Exámenes

El generador de exámenes es una clase que escribe directamente sobre un archivo cada línea de la estructura elegida del documento VXML consistente en examen de opción múltiple. El generador reutiliza código de otra clase (VxmlGenerator), con el objetivo de reducir considerablemente la escritura del documento VXML. La generación de los exámenes de opción múltiple fue seleccionada de esta forma, debido a que las secciones CDATA, muy importantes para este recurso de aprendizaje, no pudiesen haber sido agregadas de otra forma (por ejemplo si se hubiera hecho la generación con XSLT, la librería tomaría en cuenta esta sección como parte de las instrucciones de transformación y no la imprimiría tal cual en el documento transformado consistente en el examen).

```
package generator;

import java.io.File;
import java.io.PrintWriter;
import java.util.List;
import org.apache.commons.logging.Log;
import org.apache.commons.logging.LogFactory;
import vxmlgenerator.domain.Question;

public class QuizVxmlGenerator extends VxmlGenerator {

    /** Logger for this class and subclasses */
    protected final Log logger = LogFactory.getLog(getClass());

    /** creating a vxml quiz type exercise */
    public void createQuiz(List<Question> questions,
        String topic, int exerciseId, String filename) {
        try{
            File file = new File(filename); // vxmldocs/quiz.xml
            if(!file.exists()){
                file.createNewFile();
            }

            PrintWriter out = new PrintWriter(file);
            StringBuilder vxml = new StringBuilder();

            //Header and welcome menu adding
            vxml.append(XMLHEADER);
            vxml.append(VXMLHEADER);

            vxml.append(MENU);
            vxml.append(addPromptEnum("<break/>Welcome to your " + topic + "
            quiz. Please say <break/>"));
            vxml.append(addChoice("#quiz", "Start"));
            vxml.append(MENUEND);

            vxml.append(addForm("quiz"));
            StringBuilder namelist = new StringBuilder();
```

```

        for (Question question: questions) {
            int i = question.getQuestionNumber();
            vxml.append(addField("question"+i));
            vxml.append(PROMPTNOBARGEIN);

            vxml.append("<break/>" + question.getQuestion() + "<break/>\n");
            vxml.append("1 <break/>" + question.getAnswer1() + "<break/>\n");
            vxml.append("2 <break/>" + question.getAnswer2() + "<break/>\n");
            vxml.append("3 <break/>" + question.getAnswer3() + "<break/>\n");
            vxml.append(PROMPTEND);

            vxml.append(GRAMARVOICE);
            vxml.append("<![CDATA[" + "\n");
            vxml.append("[" + "\n");
            vxml.append("[(1) (it is 1)] {<question"+i+ " \"1\">}"+ "\n");
            vxml.append("[(2) (it is 2)] {<question"+i+ " \"2\">}"+ "\n");
            vxml.append("[(3) (it is 3)] {<question"+i+ " \"3\">}"+ "\n");
            vxml.append("]" + "\n");
            vxml.append("]]>" + "\n");
            vxml.append(GRAMAREND);

            vxml.append(addNoInput("Not getting any input"));
            vxml.append(addNoMatch("We got 1, 2 or 3. Try again"));
            vxml.append(FIELDEND);

            namelist.append("question"+i+ " ");
        }

//adding student id field
vxml.append(addField("studentid"));
vxml.append(GRAMARDTMF);
vxml.append(addPrompt("Use your telephone keypad to enter your
Student id followed by the number key.));
vxml.append(FIELDEND);

//adding exercise id variable
vxml.append("<var name=\"exerciseid\" expr=\""+exerciseId+"\"/>");

namelist.append("studentid exerciseid");
vxml.append(addFilledSubmit(
"http://cs.mty.itesm.mx:82/vxmlgenerator/quizervlet.htm",
namelist.toString()));

//closing vxml file
vxml.append(FORMEND);
vxml.append(VXMLEND);

//flush file
out.print(vxml);
out.flush();

} catch (Throwable t) {
}
}
}

```

Anexo Generador General de Exámenes

El generador general de exámenes es una clase con constantes y métodos que son reutilizados por la clase de generación de exámenes. El objetivo de esta clase es disminuir la cantidad de código para la creación del documento VXML consistente en examen de opción múltiple.

```
package generator;

public abstract class VxmlGenerator {

    protected final String MENU = "<menu>\n";
    protected final String MENUEND = "</menu>\n";
    protected final String PROMPT = "<prompt>\n";
    protected final String PROMPTNOBARGEIN = "<prompt
    bargein=\"false\">\n";
    protected final String PROMPTEND = "</prompt>\n";
    protected final String FIELD = "<field>\n";
    protected final String FIELDEND = "</field>\n";
    protected final String FORM = "<form>\n";
    protected final String FORMEND = "</form>\n";
    protected final String FILLED = "<filled>\n";
    protected final String FILLEDEND = "</filled>\n";
    protected final String ENUMERATE = "<enumerate/>\n";

    protected final String GRAMARVOICE = "<grammar type=\"application/x-
    gsl\" mode=\"voice\">\n";
    protected final String GRAMARDTMF = "<grammar
    src=\"builtin:dtmf/digits\" mode=\"dtmf\"/>\n";
    protected final String GRAMAREND = "</grammar>\n";

    protected final String XMLHEADER = "<?xml version=\"1.0\"
    encoding=\"iso-8859-1\"?>\n";
    protected final String VXMLHEADER = "<vxml version=\"2.0\"
    xmlns=\"http://www.w3.org/2001/vxml\" xml:lang=\"en-us\">\n";
    protected final String VXMLEND = "</vxml>\n";

    protected String addChoice(String next, String message){
        StringBuilder choice = new StringBuilder();
        choice.append("<choice next=\"\"");
        choice.append(next);
        choice.append("\>");
        choice.append(message);
        choice.append("</choice>\n");
        return choice.toString();
    }

    protected String addPromptEnum(String message) {
        StringBuilder prompt = new StringBuilder();
        prompt.append(PROMPT);
        prompt.append(message);
        prompt.append(ENUMERATE);
        prompt.append(PROMPTEND);
        return prompt.toString();
    }
}
```

```

    }

    protected String addForm(String id){
        StringBuilder form = new StringBuilder();
        form.append("<form id=\"");
        form.append(id);
        form.append("\"><n");
        return form.toString();
    }

    protected String addField(String name) {
        StringBuilder field = new StringBuilder();
        field.append("<field name=\"");
        field.append(name);
        field.append("\"><n");
        return field.toString();
    }

    protected String addNoInput(String message) {
        StringBuilder noInput = new StringBuilder();
        noInput.append("<noinput>");
        noInput.append(message);
        noInput.append("</noinput><n");
        return noInput.toString();
    }

    protected String addNoMatch(String message) {
        StringBuilder noMatch = new StringBuilder();
        noMatch.append("<nomatch>");
        noMatch.append(message);
        noMatch.append("</nomatch><n");
        return noMatch.toString();
    }

    protected String addPrompt(String message) {
        StringBuilder prompt = new StringBuilder();
        prompt.append(PROMPT);
        prompt.append(message);
        prompt.append(PROMPTEND);
        return prompt.toString();
    }

    protected String addFilledSubmit(String next, String namelist) {
        StringBuilder filled = new StringBuilder();
        filled.append(FILLED);
        filled.append("<submit next=\"");
        filled.append(next);
        filled.append("\ " namelist="");
        filled.append(namelist);
        filled.append("\ "/><n");
        filled.append(FILLEDEND);
        return filled.toString();
    }

    protected String addFilledSubmitAudio(String next, String namelist) {
        StringBuilder filled = new StringBuilder();

```

```
        filled.append(FILLED);
        filled.append("<submit next=\"");
        filled.append(next);
        filled.append("\" method=\"post\" ");
        filled.append("enctype=\"multipart/form-data\" ");
        filled.append("namelist=\"");
        filled.append(namelist);
        filled.append("\"/>\n");
        filled.append(FILLEDEND);
        return filled.toString();
    }

    protected String addRecord(String recordName, String prompt){
        StringBuilder record = new StringBuilder();
        record.append("<record name=\"");
        record.append(recordName);
        record.append("\" beep=\"true\" dtmfterm=\"true\"
type=\"audio/basic\">\n");
        record.append(PROMPT);
        record.append(prompt);
        record.append(PROMPTEND);
        record.append("</record>\n");
        return record.toString();
    }
}
```

Anexo Generador de datos de Exámenes orales, Blog y Anuncios

El XmlGenerator es una clase cuya función es la creación o modificación de archivos XML que contienen información de las aplicaciones de voz consistentes en los recursos de aprendizaje de examen oral y blog, adicionalmente de datos de la aplicación de voz consistente de anuncios. La creación de archivos XML se realiza mediante la impresión línea por línea del archivo, mientras que la modificación o actualización se realiza gracias a la librería Dom4j, a través de la creación de nuevos nodos de los elementos del XML.

```
package generator;

import java.io.File;
import java.io.PrintWriter;
import java.io.StringWriter;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.List;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.transform.OutputKeys;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;
import org.w3c.dom.*;
import org.xml.sax.SAXException;
import org.xml.sax.SAXParseException;
import vxmlgenerator.domain.Announcement;
import vxmlgenerator.domain.Question;

public class XmlGenerator {

    protected final String LB = "\n";

    public void createQuizData(List<Question> questions,
String topic, int exerciseId, String filename){
        try{
            File file = new File(filename); // vxmldocs/quizdata.xml
            if(!file.exists()){
                file.createNewFile();
            }

            PrintWriter out = new PrintWriter(file);

            StringBuilder namelist = new StringBuilder();
            StringBuilder xml = new StringBuilder();

            xml.append("<?xml version=\"1.0\" encoding=\"ISO-8859-1\"?>").append(LB);
            xml.append("<questions>").append(LB);
            xml.append("
<exerciseid>"+exerciseId+"</exerciseid>").append(LB);
            xml.append("
    <topic>"+topic+"</topic>").append(LB);
```

```

        for(Question question: questions){
            xml.append("        <question>").append(LB);
            xml.append("
<name>question"+question.getQuestionNumber()+"</name>").append(LB);
            xml.append("                <description>"+
question.getQuestion()+"</description>").append(LB);
            xml.append("                <answer1>"+
question.getAnswer1()+"</answer1>").append(LB);
            xml.append("                <answer2>"+
question.getAnswer2()+"</answer2>").append(LB);
            xml.append("                <answer3>"+
question.getAnswer3()+"</answer3>").append(LB);
            xml.append("                <correctanswer>"+
question.getCorrectAnswer()+"</correctanswer>").append(LB);
            xml.append("        </question>").append(LB);
            namelist.append("question"+question.getQuestionNumber()
+" ");
        }

        namelist.append("studentid");
        xml.append("        <namelist>"+namelist+"</namelist>").append(LB);
        xml.append("</questions>");
        //flush file
        out.print(xml);
        out.flush();
        out.close();

    }catch (Throwable t) {
        System.out.println(t.getMessage());
    }
}

/**
 *
 * @param questions
 * @param topic
 */
public void createOralQuizData(List<Question> questions,
String topic, int exerciseId, String filename){
    try{
        File file = new File(filename);
        if(!file.exists()){
            file.createNewFile();
        }

        PrintWriter out = new PrintWriter(file);

        StringBuilder namelist = new StringBuilder();
        StringBuilder xml = new StringBuilder();

        xml.append("<?xml version=\"1.0\" encoding=\"ISO-8859-
1\"?>").append(LB);
        xml.append("<questions>").append(LB);
        xml.append("
<exerciseid>"+exerciseId+"</exerciseid>").append(LB);

```

```

        xml.append("        <topic>"+topic+"</topic>").append(LB);

        for(Question question: questions){
            xml.append("        <question>").append(LB);
            xml.append("
<name>question"+question.getQuestionNumber()+"</name>").append(LB);
            xml.append("                <description>"+
            question.getQuestion()+"</description>").append(LB);
            xml.append("        </question>").append(LB);
            namelist.append("question"+question.getQuestionNumber()
            +" ");
        }

        xml.append("        <namelist>"+namelist+"</namelist>").append(LB);
        xml.append("</questions>");
        //flush file
        out.print(xml);
        out.flush();
        out.close();

    }catch (Throwable t) {
        System.out.println(t.getMessage());
    }
}

/**
 *
 * @param announcements
 */
public void createAnnouncementsData(List<Announcement> announcements,
String filename){

try{
    File file = new File(filename);
    // vxmldocs/announcementsdata.xml
    if(!file.exists()){
        file.createNewFile();
    }

    PrintWriter out = new PrintWriter(file);

    DateFormat formatter = new SimpleDateFormat("EEEE MMMM dd yyyy");
    StringBuilder date = new StringBuilder();
    StringBuilder xml = new StringBuilder();

    xml.append("<?xml version=\"1.0\" encoding=\"ISO-8859-
1\"?>").append(LB);
    xml.append("<announcements>").append(LB);

    for(Announcement announcement: announcements){
        date.append(formatter.format(announcement.getDate()));
        xml.append("        <announcement>").append(LB);
        xml.append("                <description><![CDATA[";
        xml.append(announcement.getAnnouncement());
        xml.append("]]></description>").append(LB);
        xml.append("                <date>"+

```

```

        date.toString()+"</date>").append(LB);
        xml.append("        </announcement>").append(LB);
        date.delete(0,date.length());
    }

    xml.append("</announcements>");

    //flush file
    out.print(xml);
    out.flush();
    out.close();

}catch (Throwable t) {
    System.out.println(t.getMessage());
}
}

/**
 *
 * @param audioFilename
 * @param xmlFilename
 */
public void addBlogData(String audioFilename, String xmlFilename){
    try {

        File xmlDataFile = new File(xmlFilename);

        if(xmlDataFile.exists()){
            DocumentBuilderFactory docBuilderFactory =
                DocumentBuilderFactory.newInstance();
            DocumentBuilder docBuilder =
                docBuilderFactory.newDocumentBuilder();
            Document xmlDoc = docBuilder.parse (new
                File(xmlFilename));

            // normalize text representation
            xmlDoc.getDocumentElement ().normalize ();

            Node blogNode = xmlDoc.getFirstChild();
            Node filepath = xmlDoc.createElement("filepath");
            Node path = xmlDoc.createElement("path");
            path.setTextContent(audioFilename);
            filepath.appendChild(path);
            blogNode.appendChild(filepath);

            Transformer transformer =
                TransformerFactory.newInstance().newTransformer();
            transformer.setOutputProperty(OutputKeys.INDENT,
                "yes");

            //initialize StreamResult with File object to save to
            //file
            StreamResult result = new StreamResult(new
                StringWriter());
            DOMSource source = new DOMSource(xmlDoc);
            transformer.transform(source, result);

```

```

        String xmlString = result.getWriter().toString();

        PrintWriter out = new PrintWriter(xmlDataFile);
        out.print(xmlString);
        out.flush();
        out.close();

    }
} catch (SAXParseException err) {
    System.out.println ("** Parsing error" + ", line " +
        err.getLineNumber () + ", uri " + err.getSystemId ());
    System.out.println(" " + err.getMessage ());
} catch (SAXException e) {
}

/**
 *
 * @param audioFilename
 * @param xmlFilename
 */
public void createBlogData(String xmlFilename,
    String blogAbout, int exerciseId, String audioFilename){

    try {

        File xmlDataFile = new File(xmlFilename);

        //Creating xml data file
        xmlDataFile.createNewFile();

        PrintWriter out = new PrintWriter(xmlDataFile);
        StringBuilder xml = new StringBuilder();

        xml.append("<?xml version=\"1.0\" encoding=\"ISO-8859-
1\"?>").append(LB);
        xml.append("<blog>").append(LB);
        xml.append("<blogabout>"+blogAbout+"</blogabout>").append(LB);
        xml.append("<exerciseid>"+exerciseId+"</exerciseid>").append(LB);
        xml.append("<filepath>"+audioFilename+"</filepath>").append(LB);
        xml.append("</blog>").append(LB);

        //flush file
        out.print(xml);
        out.flush();
        out.close();

    } catch (Exception e) {
        e.printStackTrace();
    }
}
}

```

Anexo Documento VXML consistente en Examen

El siguiente archivo VXML es la estructura del documento VXML elegido para que una vez ejecutado en el servidor de voz funcione como una aplicación de voz consistente en un examen de opción múltiple. Las secciones CDATA ayudan como mecanismo de manejo de errores de usuario, haciendo que posibles respuestas del alumno se asocien a una sola. Los elementos <noinput> y <nomatch> sirven como mecanismos de manejo de errores del usuario cuando este responde de manera extraña o distinta a lo que se espera.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<vxml version="2.0" xmlns="http://www.w3.org/2001/vxml" xml:lang="en-us">
<menu>
<prompt>
<break/>Welcome to your example quiz. Please say <break/><enumerate/>
</prompt>
<choice next="#quiz">Start</choice>
</menu>
<form id="quiz">
<field name="question1">
<prompt bargein="false">
<break/>what is a computer<break/>
1 <break/>a machine<break/>
2 <break/>a person<break/>
3 <break/>a thing<break/>
</prompt>
<grammar type="application/x-gsl" mode="voice">
<![CDATA[
[
[(1) (it is 1)] {<question1 "1">}
[(2) (it is 2)] {<question1 "2">}
[(3) (it is 3)] {<question1 "3">}
]
]]>
</grammar>
<noinput>Not getting any input</noinput>
<nomatch>We got 1, 2 or 3. Try again</nomatch>
</field>
<field name="studentid">
<grammar src="builtin:dtmf/digits" mode="dtmf"/>
<prompt>
Use your telephone keypad to enter your Student i d followed by the number
key.</prompt>
</field>
<var name="exerciseId" expr="4"/><filled>
<submit next="http://cs.mty.itesm.mx:82/vxmlgenerator/quizservlet.htm"
namelist="question1 studentid exerciseid"/>
</filled>
</form>
</vxml>
```

Anexo Plantilla XSL para creación de Examen oral

El siguiente archivo XSL sirve para el establecimiento de la estructura del documento VXML consistente en examen oral, con el cual es posible mantener separados los datos (preguntas y respuestas) del recurso de aprendizaje en un archivo XML. Lo mencionado permite que en dado caso de realizar un cambio en la estructura no comprometa ninguna modificación de código de la aplicación web, lo que se traduce en una mayor mantenibilidad para este tipo de recurso. En el archivo se observa que un atributo del elemento <record> es el de “maxtime”, el cual es utilizado para evitar que el alumno se prolongue en su respuesta.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet                                version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="xml" omit-xml-declaration="yes" />
<xsl:template match="/">

<vxml version="2.1">
<prompt>You have 30 seconds to answer each question or you could press any
key if you are finished.<break size="large"/></prompt>
<form>
<xsl:for-each select="questions/question">
<xsl:element name="record">
<xsl:attribute name="name"><xsl:value-of select="name"/></xsl:attribute>
<xsl:attribute name="maxtime">30s</xsl:attribute>
<xsl:attribute name="finalsilence">5s</xsl:attribute>
<xsl:attribute name="beep">true</xsl:attribute>
<xsl:attribute name="dtmfterm">true</xsl:attribute>

<prompt bargein="false"><break/><xsl:value-of select="description"/><break/>
</prompt>

</xsl:element>
</xsl:for-each>

<filled>
<xsl:element name="submit">
<xsl:attribute
name="next">http://cs.mty.itesm.mx:82/vxmlgenerator/oralquizervlet.htm</xsl:
attribute>
<xsl:attribute                                name="namelist"><xsl:value-of
select="questions/namelist"/></xsl:attribute>
<xsl:attribute name="enctype">multipart/form-data</xsl:attribute>
<xsl:attribute name="method">post</xsl:attribute>
</xsl:element>
</filled>

</form>
</vxml>
</xsl:template>
</xsl:stylesheet>
```

Anexo Plantilla XSL para creación de Blog

El siguiente archivo XSL sirve como estructura del documento VXML consistente en el blog, con el cual es posible mantener separados los datos (aportaciones) del recurso de aprendizaje en un archivo XML. Con lo anterior en caso de necesitar un cambio en la estructura solo se realizaría sobre el archivo XSL y no en código. En el archivo se observa que un atributo del elemento <record>, en la forma de contribución (contributionForm) es el de “maxtime”, el cual es utilizado para evitar que el alumno se prolongue en su contribución. Se utiliza el elemento <link> para darle oportunidad al alumno de regresar al menú principal en caso de que utilice las palabras de “stop”, “return” o “menú”.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet                                version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="xml" omit-xml-declaration="yes" />
<xsl:template match="/">
<vxml version="2.0" xmlns="http://www.w3.org/2001/vxml" xml:lang="en-us">

<link next="#Menu">
  <grammar root="main">
    <rule id="main" scope="public">
      <one-of>
        <item> stop </item>
        <item> return </item>
        <item> menu </item>
      </one-of>
    </rule>
  </grammar>
</link>

<form id="Menu">

<block><prompt bargein="true">Welcome to the school blog where you can hear
other students contributions or make one of your own<br/>
</prompt></block>

  <field name="makeContribution" type="boolean" modal="true">
    <prompt bargein="false">Do you want to make a contribution?</prompt>
    <filled>
    <if cond="makeContribution == true"><goto next="#contributionForm"/>
    <else/><goto nextitem="hearContribution"/>
    </if>
    </filled>
  </field>

  <field name="hearContribution" type="boolean" modal="true">
    <prompt bargein="false">Do you want hear contributions?</prompt>
    <filled>
    <if cond="hearContribution == true">
    <goto next="#hearContributionsForm"/>
    <else/><goto nextitem="noAction"/>
    </if>
```

```

        </filled>
    </field>
    <block name="noAction">
    <prompt>
        We will now return you to the main menu.
        Or, you may hang up, if you are finished.
    </prompt>
    <goto next="#Menu"/>
    </block>
</form>

<form id="hearContributionsForm">
    <block>
    <prompt>
    <audio src="content.wav">
    <break/>To make a contribution,
    you can say main menu at any time.<break/>
    </audio>
    <break size="medium"/>
    </prompt>

    <xsl:for-each select="blog/filepath">
    <xsl:element name="audio">
    <xsl:attribute name="src"><xsl:value-of select="path"/></xsl:attribute>
    </xsl:element>
    </xsl:for-each>

    <prompt bargein="true">
    <audio src="noMore.wav">
    <break/>There are no more contributions available.
    </audio>
    <audio src="returnToMain.wav">
    <break/>We will now return you to the main menu.
    Or, you may hang up, if you are finished.<break/>
    </audio>
    </prompt>

    <goto next="#Menu"/>
    </block>
</form>

<form id="contributionForm">
<record name="blogContribution" maxtime="30s" finalsilence="5s" beep="true"
dtmfterm="true">
    <prompt bargein="false">
    <xsl:value-of select="blog/blogabout"/><break/>
    </prompt>
    </record>
<filled>
<submit          next="http://cs.mty.itesm.mx:82/vxmlgenerator/blogervlet.htm"
method="post" enctype="multipart/form-data" namelist="blogContribution"/>
</filled>
</form>
</vxml>
</xsl:template>
</xsl:stylesheet>

```

Anexo Documento XML con datos de Blog

Archivo XML con información referente a la ubicación en el servidor web de los archivos de audio conteniendo aportaciones de alumnos. Cada archivo cuenta con el identificador del ejercicio en su nombre. El archivo es actualizado cada vez que se realiza una aportación.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<blog>
<blogabout>what was chapter 5 about</blogabout>
<exerciseid>2</exerciseid>
<filepath>C:\Users\cesar\Desktop\Workspace\.metadata\.plugins\org.eclipse.wst
.server.core\tmp0\wtpwebapps\vxmllgenerator\vxmlaudios\exid2-sid-
10001.wav</filepath>
</blog>
```

Anexo Documento XML con datos de Anuncios

Archivo XML con información referente a anuncios agregados por el profesor. El archivo es actualizado cada vez que el profesor agrega un nuevo anuncio.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<announcements>
  <announcement>
    <description><![CDATA[read chapter 5]]></description>
    <date>Friday April 01 2011</date>
  </announcement>
  <announcement>
    <description><![CDATA[read chapter 6]]></description>
    <date>Monday April 04 2011</date>
  </announcement>
</announcements>
```

Anexo Documento XML con datos de Examen oral

Archivo XML con información referente a preguntas y respuestas del examen oral. El archivo se crea cada vez que se crea un nuevo examen oral.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<questions>
  <exerciseid>5</exerciseid>
  <topic>computers</topic>
  <question>
    <name>question1</name>
    <description>what is a computer</description>
  </question>
  <namelist>question1 studentid exerciseid</namelist>
</questions>
```

Anexo Documento XML con datos de Blog

Archivo XML con información referente a la ubicación en el servidor web de los archivos de audio conteniendo aportaciones de alumnos. Cada archivo cuenta con el identificador del ejercicio en su nombre. El archivo es actualizado cada vez que se realiza una aportación.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<blog>
<blogabout>what was chapter 5 about</blogabout>
<exerciseid>2</exerciseid>
<filepath>C:\Users\cesar\Desktop\Workspace\.metadata\.plugins\org.eclipse.wst
.server.core\tmp0\wtpwebapps\vxmllgenerator\vxmlaudios\exid2-sid-
10001.wav</filepath>
</blog>
```

Anexo Documento XML con datos de Anuncios

Archivo XML con información referente a anuncios agregados por el profesor. El archivo es actualizado cada vez que el profesor agrega un nuevo anuncio.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<announcements>
  <announcement>
    <description><![CDATA[read chapter 5]]></description>
    <date>Friday April 01 2011</date>
  </announcement>
  <announcement>
    <description><![CDATA[read chapter 6]]></description>
    <date>Monday April 04 2011</date>
  </announcement>
</announcements>
```

Anexo Documento XML con datos de Examen oral

Archivo XML con información referente a preguntas y respuestas del examen oral. El archivo se crea cada vez que se crea un nuevo examen oral.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<questions>
  <exerciseid>5</exerciseid>
  <topic>computers</topic>
  <question>
    <name>question1</name>
    <description>what is a computer</description>
  </question>
  <namelist>question1 studentid exerciseid</namelist>
</questions>
```

Anexo Documento XML con datos de Blog

Archivo XML con información referente a la ubicación en el servidor web de los archivos de audio conteniendo aportaciones de alumnos. Cada archivo cuenta con el identificador del ejercicio en su nombre. El archivo es actualizado cada vez que se realiza una aportación.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<blog>
<blogabout>what was chapter 5 about</blogabout>
<exerciseid>2</exerciseid>
<filepath>C:\Users\cesar\Desktop\Workspace\.metadata\.plugins\org.eclipse.wst
.server.core\tmp0\wtpwebapps\vxmllgenerator\vxmlaudios\exid2-sid-
10001.wav</filepath>
</blog>
```

Anexo Documento XML con datos de Anuncios

Archivo XML con información referente a anuncios agregados por el profesor. El archivo es actualizado cada vez que el profesor agrega un nuevo anuncio.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<announcements>
  <announcement>
    <description><![CDATA[read chapter 5]]></description>
    <date>Friday April 01 2011</date>
  </announcement>
  <announcement>
    <description><![CDATA[read chapter 6]]></description>
    <date>Monday April 04 2011</date>
  </announcement>
</announcements>
```

Anexo Documento XML con datos de Examen oral

Archivo XML con información referente a preguntas y respuestas del examen oral. El archivo se crea cada vez que se crea un nuevo examen oral.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<questions>
  <exerciseid>5</exerciseid>
  <topic>computers</topic>
  <question>
    <name>question1</name>
    <description>what is a computer</description>
  </question>
  <namelist>question1 studentid exerciseid</namelist>
</questions>
```

Anexo Documento VXML consistente en Examen oral

El siguiente archivo es un ejemplo de documento VXML que resulta después de la transformación, con la librería XSLT, del archivo XSL y el archivo XML de datos del examen oral. Es de importancia notar que una vez recolectadas las respuestas en formato de audio, estas son enviadas a través del protocolo http con el método de POST (especificado en el atributo method del elemento <submit>), ya que este permite enviar una mayor cantidad de datos que el método GET. También es necesario especificar que el tipo de encriptación es el de “multipart/form-data”, ya que es la convención MIME (Multipurpose Internet Mail Extensions) para el envío de archivos provenientes de una forma, con lo cual se evita perder los identificadores del elemento <record>.

```
<vxml version="2.0" xml:lang="en-us" xmlns="http://www.w3.org/2001/vxml">
<menu><prompt><break/>Please say <break/><enumerate/>
</prompt>
<choice next="#oralquiz">Start</choice>
</menu>

<form id="oralquiz">
  <record name="question1" beep="true" maxtime="30s" dtmfterm="true"
    type="audio/basic">
    <prompt bargein="false"><break/>what is a computer<break/>
    </prompt>
  </record>

  <filled>
  <submit
next="http://cs.mty.itesm.mx:82/vxmlgenerator/oralquizservlet.htm"
method="post" enctype="multipart/form-data" namelist="question1"/>
  </filled>
</form>
</vxml>
```

Anexo Documento VXML consistente en Anuncios

El siguiente es el documento VXML que contiene las instrucciones para reproducir en audio los avisos agregados por el profesor a través de la aplicación web. Código ECMAScript es utilizado para filtrar los datos de anuncios obtenidos del archivo XML accedido con el elemento <data> y asignados a una variable del documento VXML con el elemento <assign>.

```
<?xml version="1.0" encoding="UTF-8"?>
<vxml version="2.1" xmlns:voxeo="http://community.voxeo.com/xmlns/vxml">
<script><![CDATA[
function assignArray(document, tagName, element) {
    var j=(document.getElementsByTagName(element).length );
    var dataArray = new Array();
    for(var i = 0; i < j; i++) {
    try { dataArray[i]=
        document.getElementsByTagName(tagName).item(i).firstChild.data;
    }catch(e){ dataArray[i] = '' ; }
    }
    return dataArray;
}
function assignArray(document, tagName1, tagName2, element) {
    var j=(document.getElementsByTagName(element).length );
    var dataArray = new Array();
    for(var i = 0; i < j; i++) {
    try { dataArray[i]=
        document.getElementsByTagName(tagName2).item(i).firstChild.data+"."+
        document.getElementsByTagName(tagName1).item(i).firstChild.data+ ".";
    }catch(e){ dataArray[i] = '' ;}
    }
    return dataArray;
}
}]> </script>
<var name="tagDescriptionArray"/>
<var name="tagDescriptionDateArray"/>
<var name="xmlDocument"/>

<form id="announcements">
<block>
<data name="xmlDocument" srcexpr=
"'http://cs.mty.itesm.mx:82/vxmlgenerator/vxmldocs/announcementsdata.xml'"/>
<assign name="document.tagDescriptionArray" expr="assignArray(xmlDocument,
'description', 'announcement')"/>
<assign name="document.tagDescriptionDateArray"
expr="assignArray(xmlDocument, 'description', 'date', 'announcement')"/>

<foreach item="announcement" array="document.tagDescriptionDateArray">
<prompt><prosody volume="soft"><value expr="announcement"/></prosody>
<break size="large"/></prompt>
</foreach>

</block>
</form>
</vxml>
```

Anexo Servlet para evaluación de exámenes

La siguiente clase es un servlet, que recibe las respuestas de los alumnos provenientes de un examen de opción múltiple. Con el acceso a la base de datos, se realiza la comparación de respuestas del alumno y respuestas correctas para poder asignar una calificación al examen. La calificación es regresada al alumno y promediada con las demás calificaciones obtenidas de otros exámenes.

```
package vxmgenerator.servlet;

import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import org.apache.commons.logging.*;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.web.servlet.mvc.Controller;
import vxmgenerator.dao.*;
import vxmgenerator.domain.*;
import vxmgenerator.service.ServiceFinder;

public class QuizServlet extends HttpServlet implements Controller {

    private static final long serialVersionUID = 1L;
    protected final Log logger = LoggerFactory.getLog(getClass());
    private StudentAnswersDao studentAnswersDao;
    private StudentDao studentDao;
    private QuestionDao questionDao;
    private GradeDao gradeDao;

    @Override
    public ModelAndView handleRequest(HttpServletRequest request,
        HttpServletResponse response) throws Exception {
        logger.info("At QuizServlet");

        try{
            studentAnswersDao = (StudentAnswersDao)
                ServiceFinder.getContext(request).getBean("StudentAnswersDao");
            studentDao = (StudentDao)
                ServiceFinder.getContext(request).getBean("StudentDao");
            questionDao = (QuestionDao)
                ServiceFinder.getContext(request).getBean("QuestionDao");
            gradeDao = (GradeDao)
                ServiceFinder.getContext(request).getBean("GradeDao");

            int studentId =
                Integer.parseInt(request.getParameter("studentid"));
            int exerciseId =
                Integer.parseInt(request.getParameter("exerciseid"));

            if( studentDao.checkStudent( studentId ) ){

                Grade tmpGrade = gradeDao.loadGrade(exerciseId, studentId);
                if(tmpGrade != null){
```

```

        PrintWriter out = response.getWriter();
        out.println("<?xml version=\"1.0\"?>");
        out.println("<vxml version=\"2.0\">");

        out.println("<form>");
        out.println("    <block>");
        out.println("        <prompt>");
        out.println("    You already answered the quiz");
        out.println("        </prompt>");
        out.println("    </block>");
        out.println("</form>");
        out.println("</vxml>");
    }else{
        List<StudentAnswer> studentAnswers = new
        ArrayList<StudentAnswer>();
        List<Question> questions =
        questionDao.loadQuestions(exerciseId);
        logger.info("Questions loaded");

        double counter = 0;
        double hits = 0;

        for(int i=1; i<6; i++){

            if(request.getParameter("question"+i)!=null){
                StudentAnswer studentAnswer = new StudentAnswer();
                studentAnswer.setStudentId(studentId);
                studentAnswer.setExerciseId(exerciseId);
                studentAnswer.setQuestionNumber(i);
                studentAnswer.setExerciseDate(new Date());
                studentAnswer.setAnswer(request.getParameter("question"+i));
                studentAnswers.add(studentAnswer);
                counter++;

                int correctAnswer=0;
                for(Question question: questions){

                    if(question.getQuestionNumber()==i){
                        correctAnswer = Integer.parseInt(question.getCorrectAnswer());
                        int answer = Integer.parseInt(studentAnswer.getAnswer());
                        if(correctAnswer == answer ){
                            hits++;
                        }
                    }
                }
            }
        }

        //save student answers and exercise grade
        logger.info("About to add student answers");

        studentAnswersDao.addStudentAnswers(studentAnswers);

        Grade grade = new Grade();
        grade.setExerciseDate(new Date());

```

```

grade.setExerciseId(exerciseId);
grade.setStudentId(studentId);
grade.setGrade( (int) Math.floor((hits/counter)*10) );
gradeDao.addGrade(grade);

//update student information
logger.info("About to update GPA");
studentDao.updateGPA(studentId, (int) Math.floor((hits/counter)*10) );

logger.info("Going out from QuizServlet");
PrintWriter out = response.getWriter();
out.println("<?xml version=\"1.0\"?>");
out.println("<vxml version=\"2.0\">");

out.println("<form>");
out.println("    <block>");
out.println("        <prompt>");
out.println("Thanks, Your answers have been saved");
out.println("    You got " + grade.getGrade());
out.println("        </prompt>");
out.println("    </block>");
out.println("</form>");
out.println("</vxml>");
}
} else{
    logger.info("Student does not exist");
    PrintWriter out = response.getWriter();
    out.println("<?xml version=\"1.0\"?>");
    out.println("<vxml version=\"2.0\">");
    out.println("<form>");
    out.println("    <block>");
    out.println("        <prompt>");
    out.println("        Incorrect student id.");
    out.println("        </prompt>");
    out.println("    </block>");
    out.println("</form>");
    out.println("</vxml>");
}
} catch(Exception e){
    PrintWriter out = response.getWriter();
    out.println("<?xml version=\"1.0\"?>");
    out.println("<vxml version=\"2.0\">");
    out.println("<form>");
    out.println("    <block>");
    out.println("        <prompt>");
    out.println("An error in the server occurred");
    out.println("        </prompt>");
    out.println("    </block>");
    out.println("</form>");
    out.println("</vxml>");
}
}
return null;
}
}
package vxmlgenerator.servlet;

```

Anexo Servlet para salvado de respuestas orales

La siguiente clase es un servlet, que recibe las respuestas de los alumnos en formato de audio provenientes de un examen oral. Los archivos de audio vienen embebidos en el “request” y son obtenidos a través del parseo de dicho objeto con la clase “ServletFileUpload” de la librería “commons-fileupload”. La función principal del servlet es asignarle un nombre a cada archivo de audio para que a través de éste, se pueda identificar de qué pregunta, ejercicio y alumno proviene.

```
import java.io.*;
import java.text.*;
import java.util.*;
import javax.servlet.http.*;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.FileItemFactory;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;
import org.apache.commons.logging.*;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.web.servlet.mvc.*;
import org.w3c.dom.Document;
import org.w3c.dom.Node;

public class OralQuizServlet extends SimpleFormController implements
Controller{

    private static final long serialVersionUID = 7881059391774338638L;
    protected final Log logger = LoggerFactory.getLog(getClass());

    public ModelAndView handleRequest(HttpServletRequest request,
HttpServletResponse response) throws Exception {

        logger.info("At Oral Quiz Servlet");
        PrintWriter out = response.getWriter();
        int studentId = Integer.parseInt(request.getParameter("studentid"));

        if(ServletFileUpload.isMultipartContent(request)){
            logger.info("Request is multipart content");

            DateFormat dateFormat = new SimpleDateFormat("yyyy_MM-dd_HH-mm-ss");
            Date date = new Date();
            String dateString = dateFormat.format(date);

            FileItemFactory factory = new DiskFileItemFactory();
            ServletFileUpload upload = new ServletFileUpload(factory);
            List<FileItem> fileItems = null;

            try {
```

```

logger.info("About to get audios");
fileItems = upload.parseRequest(request);
for(FileItem item : fileItems){
String exerciseId = getExerciseId();
String filename = "exid-"+ exerciseId + " sid-"+studentId + " " +

item.getFieldName() + " on " + dateString + ".au";

String inputFilePath =
getServletContext().getRealPath("vxmlaudios/"+filename);
logger.info("Audio file path:" + inputFilePath);
File file = new File(inputFilePath);
item.write(file);
}
} catch (Exception e) {
logger.info("Error message: "+e.toString());
out.println("<?xml version=\"1.0\"?>");
out.println("<vxml version=\"2.0\">");
out.println("<form>");
out.println("    <block>");
out.println("        <prompt>");
out.println("            An error in the server occurred.");
out.println("        </prompt>");
out.println("    </block>");
out.println("</form>");
out.println("</vxml>");
}

out.println("<?xml version=\"1.0\"?>");
out.println("<vxml version=\"2.0\">");
out.println("<form>");
out.println("    <block>");
out.println("        <prompt>");
out.println("            Thanks, Your answers have been saved");
out.println("        </prompt>");
out.println("    </block>");
out.println("</form>");
out.println("</vxml>");

}else{

try{
logger.info("Request is NOT multipart content");
String xsltFilename =
getServletContext().getRealPath("vxmldocs/oralquizxslt.xml");
File xmlDataFile = new File(xsltFilename);

DocumentBuilderFactory docBuilderFactory =
DocumentBuilderFactory.newInstance();
DocumentBuilder docBuilder = docBuilderFactory.newDocumentBuilder();
Document xmlDoc = docBuilder.parse(xmlDataFile);

// normalize text representation
xmlDoc.getDocumentElement().normalize();

```

```

DOMSource domSource = new DOMSource(xmlDoc);
StringWriter writer = new StringWriter();
StreamResult result = new StreamResult(writer);
TransformerFactory tf = TransformerFactory.newInstance();
Transformer transformer = tf.newTransformer();
transformer.transform(domSource, result);

StringBuffer stringXmlFile = new StringBuffer(writer.toString());
int offset = stringXmlFile.indexOf("oralquizservlet.htm")+19;
stringXmlFile.insert(offset, "?studentid="+studentId);

        //return oral quiz vxml document
        out.println(stringXmlFile.toString());

} catch (Exception e) {
    logger.info("Error message: "+e.toString());

    out.println("<?xml version=\"1.0\"?>");
    out.println("<vxml version=\"2.0\">");
    out.println("<form>");
    out.println("    <block>");
    out.println("        <prompt>");
    out.println("            An error in the server occurred.");
    out.println("        </prompt>");
    out.println("    </block>");
    out.println("</form>");
    out.println("</vxml>");
}
}

return null;
}

private String getExerciseId() {
    try {
        String xmlFilePath =
getServletContext().getRealPath("vxml/docs/oralquizdata.xml");
        File xmlFile = new File(xmlFilePath);

        DocumentBuilderFactory docBuilderFactory =
DocumentBuilderFactory.newInstance();
        DocumentBuilder docBuilder =
DocBuilderFactory.newDocumentBuilder();
        Document xmlDoc = docBuilder.parse(xmlFile);
        xmlDoc.getDocumentElement().normalize();

        Node nodeExId = xmlDoc.getElementsByTagName("exerciseid").item(0);
        return nodeExId.getTextContent();

    } catch (Exception e) {
        logger.info(e.toString());
    }

    return null;
}
}

```

Anexo Servlet para salvado de aportaciones

La siguiente clase es un servlet, que recibe las aportaciones de los alumnos en formato de audio provenientes del blog. Cada archivo de audio de cada aportación, viene embebido en el “request” y es obtenido a través del parseo de dicho objeto con la clase “ServletFileUpload” de la librería “commons-fileupload”. La función principal del servlet es asignarle un nombre a cada archivo de audio para que a través de éste, se pueda identificar en qué fecha, de qué alumno y ejercicio de blog proviene.

```
package vxmlgenerator.servlet;

import generator.*;
import java.io.*;
import java.text.*;
import java.util.*;
import javax.servlet.http.*;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.FileItemFactory;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;
import org.apache.commons.logging.Log;
import org.apache.commons.logging.LogFactory;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.web.servlet.mvc.Controller;
import org.springframework.web.servlet.mvc.SimpleFormController;
import org.w3c.dom.Document;
import org.w3c.dom.Node;

public class BlogServlet extends SimpleFormController implements Controller{

    private static final long serialVersionUID = 4494286637977599775L;
    protected final Log logger = LogFactory.getLog(getClass());
    private XmlGenerator xmlGenerator = new XmlGenerator();
    private VxmlXslGenerator xslGenerator = new VxmlXslGenerator();

    public ModelAndView handleRequest(HttpServletRequest request,
    HttpServletResponse response) throws Exception {

        logger.info("At Blog Servlet");
        PrintWriter out = response.getWriter();
        if(ServletFileUpload.isMultipartContent(request)){
            logger.info("Request is multipart content");

            DateFormat dateFormat = new SimpleDateFormat("yyyy_MM-dd_HH-mm-ss");
            Date date = new Date();
            String dateString = dateFormat.format(date);
            FileItemFactory factory = new DiskFileItemFactory();
            ServletFileUpload upload = new ServletFileUpload(factory);
            List<FileItem> fileItems = null;

            try {
```

```

        logger.info("About to get blog contribution");
        String filename="";
        fileItems = upload.parseRequest(request);
        for(FileItem item : fileItems){
String exerciseId = getExerciseId();
filename = "exid-"+exerciseId +" blog on " + dateString + ".au";
String          inputFilePath          =
getServletContext().getRealPath("vxmlaudios/"+filename);
File file = new File(inputFilePath);
item.write(file);
}

String          xmlFilename          =
getServletContext().getRealPath("vxmldocs/blogdata.xml");
String          xslFilename          =
getServletContext().getRealPath("vxmldocs/blogxsl.xsl");
String          xsltFilename          =
getServletContext().getRealPath("vxmldocs/blogxslt.xml");

xmlGenerator.addBlogData("http://cs.mty.itesm.mx:82/vxmlgenerator/vxmla
udios/"+filename, xmlFilename);
xslGenerator.generateXslt(xslFilename, xmlFilename, xsltFilename);

} catch (Exception e) {e.printStackTrace();}

out.println("<?xml version=\"1.0\"?>");
out.println("<vxml version=\"2.0\">");
out.println("<form>");
out.println("    <block>");
out.println("        <prompt>");
out.println("            Thanks, Your contribution has been saved.");
out.println("        </prompt>");
out.println("    </block>");
out.println("</form>");
out.println("</vxml>");
}
return null;
}

private String getExerciseId() {
    try {
        String          xmlFilePath          =
getServletContext().getRealPath("vxmldocs/blogdata.xml");
File xmlFile = new File(xmlFilePath);
DocumentBuilderFactory          docBuilderFactory          =
DocumentBuilderFactory.newInstance();
DocumentBuilder docBuilder = docBuilderFactory.newDocumentBuilder();
Document xmlDoc = docBuilder.parse(xmlFile);
xmlDoc.getDocumentElement().normalize();
Node nodeExId = xmlDoc.getElementsByTagName("exerciseid").item(0);
return nodeExId.getTextContent();

    } catch (Exception e) {logger.info(e.toString());}
    return null;
}
}

```

Anexo Diagrama de flujo de evaluación de un examen

