



**UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**INTEGRACIÓN DE SECOND LIFE EN LA PLATAFORMA U-CURSOS PARA EL APOYO A LOS
PROCESOS DE ENSEÑANZA-APRENDIZAJE**

**MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN
COMPUTACIÓN**

JAIME ISRAEL BAUZÁ ASTUDILLO

**PROFESOR GUÍA:
JAVIER ROLANDO VILLANUEVA GONZÁLEZ**

**MIEMBROS DE LA COMISIÓN:
NELSON ANTRANIG BALOIAN TATARYAN
PATRICIO VICENTE POBLETE OLIVARES**

**SANTIAGO DE CHILE
AGOSTO 2009**

RESUMEN

U-Cursos es una plataforma de apoyo al desarrollo de la docencia creada por el Área de Infotecnologías (ADI) de la Facultad de Ciencias Físicas y Matemáticas (FCFM) de la Universidad de Chile. U-Cursos en su constante misión por entregar herramientas tecnológicas en el campo de la educación, necesita proporcionar una solución a la pérdida de continuidad de clases presenciales, debido entre otras razones, a los viajes que los docentes realizan a seminarios, conferencias y convenciones en el extranjero o en otras regiones del país.

Este problema fue confirmado y validado a través de un estudio realizado en la presente memoria con 183 docentes de la FCFM, en el cual también se investigó el expertise computacional y uso de tecnologías en la sala de clases. Esto entregó la base para proponer el uso de un ambiente virtual de aprendizaje, lo cual se concretó con la integración de la plataforma U-Cursos con el mundo virtual 3D de Second Life, entregando así una serie de herramientas que facilitan a los usuarios realizar una clase en el ciberespacio.

La metodología de trabajo se llevó a cabo en 4 etapas; análisis de la situación actual que abarcó un sondeo a las formas y normas de desarrollo tanto en U-Cursos como Second Life; toma de requerimientos, para establecer las metas y objetivos a cumplir; y diseño e implementación de la solución, las cuales fueron realizadas mediante la metodología iterativa e incremental de desarrollo de software.

El resultado final es un auditorio virtual dentro del mundo de Second Life, el cual está equipado con un presentador de diapositivas, sillas para los asistentes y un control de acceso para realizar clases privadas cuando se requiera. Las clases deben ser planificadas y organizadas previamente en la plataforma U-Cursos a través del servicio de clases virtuales disponible para los cursos que se imparten en la FCFM. Con esto, se logró cumplir exitosamente todos los objetivos del presente trabajo, y el producto final será puesto en producción por ADI en el semestre de Otoño del 2010 para comenzar así con su utilización e impartir las primeras clases a través de este mundo virtual.

Agradecimientos

He recorrido un largo camino para llegar a escribir estas últimas líneas de mi memoria y no ha sido fácil llegar hasta aquí, sin embargo cuando uno llega a la meta y cumple sus objetivos olvida los tramos difíciles y sólo recuerda aquellos momentos y personas que te ayudaron a hacer esos momentos un poco más dulces.

Sin duda la persona más importante para mí, es a la cual quiero dedicar esta memoria, mi querida madre Ximena Astudillo, pues gracias a ella y al infinito amor, apoyo, esfuerzo y comprensión que me entrega cada día, me ha permitido siempre seguir adelante en la persecución de mis sueños y lograr todo lo que soy en la vida. Por supuesto, también agradezco al resto de mi familia, mi padre Jaime y a mis hermanos Katherine y Cristóbal por su gran cariño y apoyo, además de la comprensión que me brindaron debido al poco tiempo que les dedique debido a mis deberes.

Quiero agradecer además a muchas personas que he conocido en mi vida y me han ayudado de alguna u otra forma a seguir adelante, con su amistad, apoyo o alguna palabra de aliento. Quisiera empezar por mi amiga Belén Rojas quien ha sido una persona muy importante en mi vida universitaria, al igual que a mis amigas de Bachillerato Paulina Cañete y Bárbara Cienfuegos que siempre han estado entregándome su amistad y alegría desde que nos conocimos. A mis amigos y compañeros de batallas de plan común; Martín Peredo, que pese a tener un camino diferente siempre me ha ayudado y apoyado en las diferentes tareas y desafíos que me han tocado realizar, y a Gabriel Petit que siempre estuvo disponible para ofrecerme su ayuda, conocimiento y apoyo desinteresado en todas mis ideas, tareas, prácticas profesionales, y en el trabajo de esta memoria. A mis amigos del Liceo Lastarria Alfonso Leiva y Claudio Escobar que siempre han estado ahí para apoyarme y brindarme su amistad incondicional. A mis amigos más recientes; Juan Vivanco por esas largas conversaciones y consejos de vida, y a Lorena Méndez con la cual he vivido muy gratos momentos con su compañía, cariño y apoyo.

También quiero dar un agradecimiento especial al Área de Infotecnologías (ADI) de la Facultad de Ciencias Físicas y Matemáticas, primero por darme la oportunidad de trabajar con ellos y confiar tanto en mi persona como en mi trabajo, pues fue el lugar donde adquirí una gran cantidad de experiencia tanto en lo laboral como en lo humano. Además agradezco a cada uno de los miembros del Área por su apoyo y comprensión con mis estudios y por su gran calidez humana tanto dentro como fuera de la oficina.

Finalmente quiero agradecer a mi profesor guía Javier Villanueva por todo el apoyo, confianza, comprensión y consejos entregados a lo largo de mi trabajo en esta memoria.

Índice de Contenidos

1	INTRODUCCIÓN	7
1.1	MOTIVACIÓN	8
1.2	OBJETIVOS	9
1.2.1	<i>Objetivos Generales</i>	9
1.2.2	<i>Objetivos Específicos</i>	9
1.3	JUSTIFICACIÓN	10
1.3.1	<i>Encuesta</i>	10
1.3.2	<i>Validación de Second Life</i>	18
2	MARCO CONCEPTUAL	25
2.1	CONCEPTOS GENERALES	25
2.2	CONCEPTOS DE SECOND LIFE	26
2.3	TECNOLOGÍAS ASOCIADAS	27
3	METODOLOGÍA	28
3.1	ANÁLISIS DE SITUACIÓN ACTUAL	28
3.2	REQUERIMIENTOS DEL SISTEMA	28
3.3	METODOLOGÍA DE DESARROLLO	28
3.3.1	<i>Aprendizaje de Interacción U-Cursos y Second Life</i>	28
3.3.2	<i>Aprendizaje Interacción, Creación de Objetos y Scripts en Second Life</i>	29
3.4	DISEÑO DEL SISTEMA	29
3.5	IMPLEMENTACIÓN DEL SISTEMA	29
4	ANÁLISIS SITUACIÓN ACTUAL	30
4.1	DESCRIPCIÓN DEL PROBLEMA	30
4.2	SECOND LIFE EN LA EDUCACIÓN	31
5	REQUERIMIENTOS DEL SISTEMA	32
5.1	PROBLEMÁTICA	32
5.2	REQUERIMIENTOS FUNCIONALES	32
5.2.1	<i>Requerimientos en U-Cursos</i>	32
5.2.2	<i>Requerimientos en Second Life</i>	33
5.2.3	<i>Requerimientos Globales</i>	33
5.3	REQUERIMIENTOS NO FUNCIONALES	34
6	DISEÑO DE LA SOLUCIÓN	35
6.1	DESCRIPCIÓN DE LA SOLUCIÓN	35
6.1.1	<i>Características de los Usuarios</i>	35
6.1.2	<i>Visión General</i>	36
6.2	DISEÑO ARQUITECTÓNICO	37
6.2.1	<i>Arquitectura Física</i>	37
6.2.2	<i>Arquitectura Lógica</i>	40
6.3	DISEÑO DETALLADO	45
6.3.1	<i>Clases Virtuales U-Cursos</i>	45

6.3.2	<i>Modelo de Datos</i>	50
6.3.3	<i>Scripts de Procesos Batch U-Cursos</i>	53
6.3.4	<i>Objetos Second Life</i>	55
7	IMPLEMENTACIÓN	62
7.1	ASPECTOS GENERALES	62
7.1.1	<i>Sistema parte U-Cursos</i>	62
7.1.2	<i>Sistema parte Second Life</i>	62
7.2	PARTE U-CURSOS	63
7.2.1	<i>MVC en U-Cursos</i>	63
7.2.2	<i>Estructura del Módulo</i>	65
7.2.3	<i>Capa de Presentación</i>	67
7.3	PARTE SECOND LIFE	74
7.3.1	<i>Sala Virtual</i>	74
7.3.2	<i>Presentador de Diapositivas Web</i>	79
7.3.3	<i>Paneles Informativos</i>	82
7.4	PARTE SECOND LIFE U-CURSOS	85
7.4.1	<i>Procesos Batch U-Cursos</i>	85
8	CONCLUSIONES	87
8.1	CONTRIBUCIONES	87
8.2	LIMITACIONES	88
8.3	TRABAJO FUTURO	89
9	BIBLIOGRAFÍA	90
10	ANEXOS	92
10.1	FORMATO DE HORARIOS DE LA FCFM	92
10.2	LIMITES DE PRIMS POR REGIÓN	92
10.3	SCRIPT DE PUERTA DE ACCESO	93
10.4	SCRIPT DE PRESENTADOR WEB	95
10.5	USO LIBRERÍA XYZZYTEXT	96

Índice de Figuras

Figura 1 Preparación Clase Virtual.....	36
Figura 2 Ingreso Clase Virtual	37
Figura 3 Arquitectura Física parte U-Cursos	38
Figura 4 Arquitectura Física parte Second Life	39
Figura 5 Arquitectura Física parte Second Life U-Cursos	40
Figura 6 Patrón MVC.....	40
Figura 7 Mantenedor Clases Virtuales	41
Figura 8 Ejemplo de script Hello World en Linden Scripting Language.....	42
Figura 9 Ejemplo Request Scripting Language	43
Figura 10 Definición Clase Virtual	45
Figura 11 Diagrama UML Clases Virtuales	46
Figura 12 Datos de Avatar Personal en Second Life.....	48
Figura 13 Modelo de Datos	50
Figura 14 Herramientas de Construcción de Second Life	55
Figura 15 Auditorio Gorbea Tarima	58
Figura 16 Auditorio Gorbea Tribuna	58
Figura 17 Prototipo Sala Virtual.....	59
Figura 18 Prototipo Presentador Web	60
Figura 19 Prototipo panel informativo.....	61
Figura 20 Avatar Nobu Xue	63
Figura 21 MVC en U-Cursos	63
Figura 22 Estructura de Carpetas	65
Figura 23 Mapa de Navegación Clases Virtuales.....	67
Figura 24 Diagramación U-Cursos	68
Figura 25 Pantalla principal Clases Virtuales	69
Figura 26 Ver Detalles Clase Virtual	70
Figura 27 Agregar Clase Virtual Abierta	71
Figura 28 Agregar Clase Virtual Privada	72
Figura 29 Editar Clase Virtual.....	73
Figura 30 Sala Virtual vista de tribuna.....	76
Figura 31 Sala Virtual vista de sillas.....	77
Figura 32 Sala Virtual vista de tarima.....	77
Figura 33 Sala Virtual vista exterior.....	78
Figura 34 Sala Virtual Puerta y Teletransportador	78
Figura 35 Flujo Presentador Web.....	80
Figura 36 Presentador Web	81
Figura 37 Panel de Control Presentador Web.....	81
Figura 38 Panel de Texto Informativo 1	83
Figura 39 Panel de Texto Informativo 2	84
Figura 40 Paneles Estáticos Second Life	85
Figura 41 Scripts de procesamiento batch	86

Figura 42 Caracteres XyzyText	96
Figura 43 Orden de nombres para un panel de 60 caracteres	96
Figura 44 Ejemplo Control de Texto con librería XyzyText.....	97

Índice de Gráficos

Gráfico 1 Población Encuestada	11
Gráfico 2 Promedio de clases no realizadas al año	12
Gráfico 3 Disposición de Profesores de Cátedra.....	13
Gráfico 4 Disposición Profesores Auxiliares.....	13
Gráfico 5 Uso de tecnología, Profesores de Cátedra	15
Gráfico 6 Uso de tecnología, Profesores Auxiliares	16
Gráfico 7 Expertise computacional, Profesores de Cátedra.....	16
Gráfico 8 Expertise computacional, Profesores Auxiliares	17
Gráfico 9 Uso de Sistemas operativos	17
Gráfico 10 Recursos Esperados.....	18

Índice de Tablas

Tabla 1 Distribución de Docentes.....	11
Tabla 2 Comparativas Second Life vs Competencias	19
Tabla 3 Usuarios del Sistema.....	35
Tabla 4 Características Clase Virtual	45
Tabla 5 Información Clases Virtuales.....	47
Tabla 6 Datos Clase Virtual.....	48
Tabla 7 Detalle Tabla Clases Virtuales	51
Tabla 8 Detalle Tabla Permisos	52
Tabla 9 Detalle Tabla Presentación	53
Tabla 10 Objetos Creados en Second Life.....	57
Tabla 11 Estructura de Carpetas.....	66
Tabla 12 Datos disponibles para agregar Clase Virtual	71
Tabla 13 Detalle Procesos Batch	86
Tabla 14 Formato de Horarios de la FCFM.....	92
Tabla 15 Límites de prims por región	92

1 Introducción

U-Cursos [1] es una herramienta infotecnológica para el apoyo al proceso de docencia que surge a fines de la década de los 90 como una efectiva solución web para organizar el material educativo de la Facultad de Ciencias Físicas y Matemáticas (FCFM) de la Universidad de Chile [2].

Desde sus inicios, los objetivos planteados por sus creadores fueron:

1. Facilitar esta tecnología a la comunidad entera.
2. Uniformar y ordenar el material en Internet.
3. Construir un archivo de conocimiento clasificado on-line a lo largo del tiempo.

La confianza depositada en este sistema lo establece como un recurso de complemento a la educación y un apoyo de calidad en técnicas de enseñanza y en formas de aprendizaje. Por esta razón, U-Cursos tiene la necesidad de facilitar mayores herramientas complementarias para sus usuarios, que se ajusten a la vanguardia de las nuevas tecnologías y que persigan solucionar problemas en actividades docentes [3][4].

Algunos de los problemas actuales en asuntos docentes lo constituyen el límite físico de asistir a una misma sala de clases, lo cual justifica el nacimiento del E-Learning. Otro problema lo constituye la falta de recursos para realizar demostraciones prácticas en el aprendizaje de contenidos, por ejemplo visualizar los efectos físicos de la aplicación de fuerzas a objetos. Ambos problemas pueden ser mermados mediante la utilización de un ambiente virtual de aprendizaje, el cual permite interactuar a los actores, independiente del lugar en el que se encuentren, pero además facilita una cantidad ilimitada de recursos virtuales para realizar demostraciones prácticas de contenidos.

El tema de la memoria radica en este interés por expandir las herramientas para la enseñanza, y realizar una integración entre U-Cursos y un Ambiente Virtual de Aprendizaje a través de la comunicación y coordinación entre las plataformas.

La elección de ambiente virtual es Second Life [5], un mundo virtual en 3D creado por la compañía norteamericana Linden Labs el año 2003. La estructura de Second Life es similar a los juegos MMORPG (Massively Multiplayer Online Role-Player Game) como World of Warcraft, Ultima Online o EverQuest, con la gran diferencia que no posee un objetivo de juego en particular, no existen ganadores ni perdedores.

Second Life fue abierto al público en el año 2003 y para acceder a él basta con descargar el cliente y registrar una cuenta de usuario de forma gratuita. Uno de los aspectos a destacar de Second Life es que ha sido ampliamente utilizado como medio de enseñanza [6][7], y en la actualidad existen decenas de universidades que tienen actividades en él, por citar algunas: Universidad de Harvard, Universidad del estado de Texas, Universidad de Stanford, Universidad de Frankfurt, Universidad de Hamburgo, etc¹.

Linden Labs destaca las cualidades de su producto en el uso para la educación[8]:

“Second Life ofrece un entorno único y flexible para los educadores interesados en la enseñanza a distancia, el trabajo cooperativo por medio del ordenador, la simulación, los estudios de nuevos medios y la formación empresarial.”

“Estudiantes y educadores de cualquier parte del mundo pueden trabajar juntos en Second Life en un aula virtual en red global. El uso de Second Life como un complemento a los entornos tradicionales del aula también ofrece nuevas oportunidades para enriquecer los planes de estudio actuales.”

1.1 Motivación

El problema que se pretende mitigar es el de no perder la continuidad de clases debido a la imposibilidad de asistir físicamente a un aula por parte de todos sus actores.

En primer lugar se tendrá la posibilidad por parte de los docentes de la Facultad de Ciencias Físicas y Matemáticas (FCFM) de realizar sus clases con normalidad a pesar de los viajes que deben realizar durante los semestres que dictan clases.

En segundo lugar los alumnos podrán optar en asistir a las clases programadas en Second Life, ya sea desde sus hogares, o desde los laboratorios de computación adaptados para este fin.

Finalmente se tiene la oportunidad única de integrar dos tecnologías a la vanguardia para crear un producto poderoso para el apoyo de los procesos de aprendizaje, donde se le entregará a los docentes nuevas herramientas para innovar en sus métodos de enseñanza.

¹ Para ver la lista completa de instituciones educacionales participantes en Second Life, visitar la dirección http://simteach.com/wiki/index.php?title=Institutions_and_Organizations_in_SL.

1.2 Objetivos

1.2.1 Objetivos Generales

Entregar las herramientas necesarias a los docentes para coordinar e impartir una sesión de clases de forma ubicua. Para ello, se provee una interfaz de comunicación entre U-Cursos y Second Life.

1.2.2 Objetivos Específicos

- Encontrar una herramienta adecuada para resolver el problema de no realizar clases presenciales por motivo de viajes de los docentes.
- Proveer herramientas de gestión y administración a los usuarios para realizar una clase en la plataforma U-Cursos.
- Preparar el ambiente virtual de aprendizaje con un conjunto de herramientas interactivas que faciliten la realización de una clase en él.
- Implementar la comunicación entre las plataformas de U-Cursos y Second Life.

1.3 Justificación

Un ambiente virtual de aprendizaje es una buena alternativa para suplir la imposibilidad de realizar clases presenciales. Uno de estos es Second Life, el cual permite impartir docencia online, con modelos que ofrecen un soporte visual y animado [9]. Gracias a su alto grado de interactividad y la presencia visual de los participantes, Second Life resulta más ágil, pues provee tanto comunicación por voz, como otros elementos multimedia para apoyar la clase (slides, videos, etc.) [10].

La complejidad del presente tema de memoria radica fundamentalmente en comunicar la plataforma U-Cursos y el mundo Virtual de Second Life para preparar todas las características y herramientas necesarias para realizar una clase en forma ubicua, tanto para los docentes como para los alumnos.

Las herramientas entregadas a los docentes deben ser lo más sencillas posibles, evitando cualquier tipo de configuración e instalación avanzada de diferentes componentes, es decir, se debe entregar una interfaz simple en su uso, para permitir así una mayor aceptación y una rápida propagación en su utilización.

1.3.1 Encuesta

Para validar el interés de los docentes en la participación de un Ambiente Virtual de Aprendizaje (AVA, de ahora en adelante), se realizó una encuesta en el semestre Primavera 2008 en el curso de Introducción al Trabajo de Título (CC69E).

La encuesta fue enviada a un total de 1491 docentes, entre Profesores de Cátedra y Profesores Auxiliares, que dictaron clases entre los semestres de Primavera 2007 y Primavera 2008 en la FCFM, logrando la respuesta de 183 de ellos, lo que corresponde a un 12,3% del total de la población.

La tasa de participación de docentes divididos en Profesores de Cátedra y Profesores Auxiliares la vemos en el Gráfico 1.

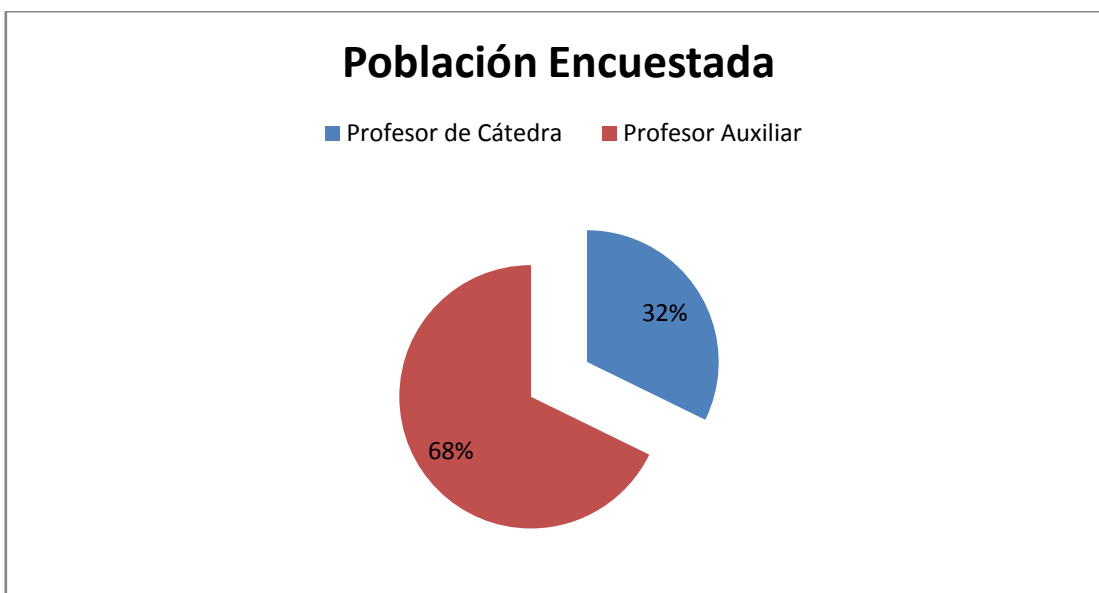


Gráfico 1 Población Encuestada

El número de cada uno de los grupos de docentes lo vemos en la Tabla 1.

Tabla 1 Distribución de Docentes

Tipo Docente	# Docentes
Profesores de Cátedra	59
Profesores Auxiliares	124
Total Encuestados	183

La encuesta cubrió diferentes temas que pretendían medir tanto la disposición a participar de los docentes en un AVA, como medir el nivel de uso y expertise de los docentes en el uso del computador. Con estos resultados se puede validar el desarrollo de la solución propuesta, asegurando su factibilidad motivacional y técnica por parte de sus principales actores.

A continuación se presentan los principales resultados.

1.3.1.1 Validación del AVA

Para validar la necesidad del sistema propuesto se consultó el número de clases no realizadas al año en promedio por los docentes por motivos de viaje. Con este resultado (que se puede ver en el Gráfico 2), se validan instancias concretas donde la solución propuesta puede ser utilizada.

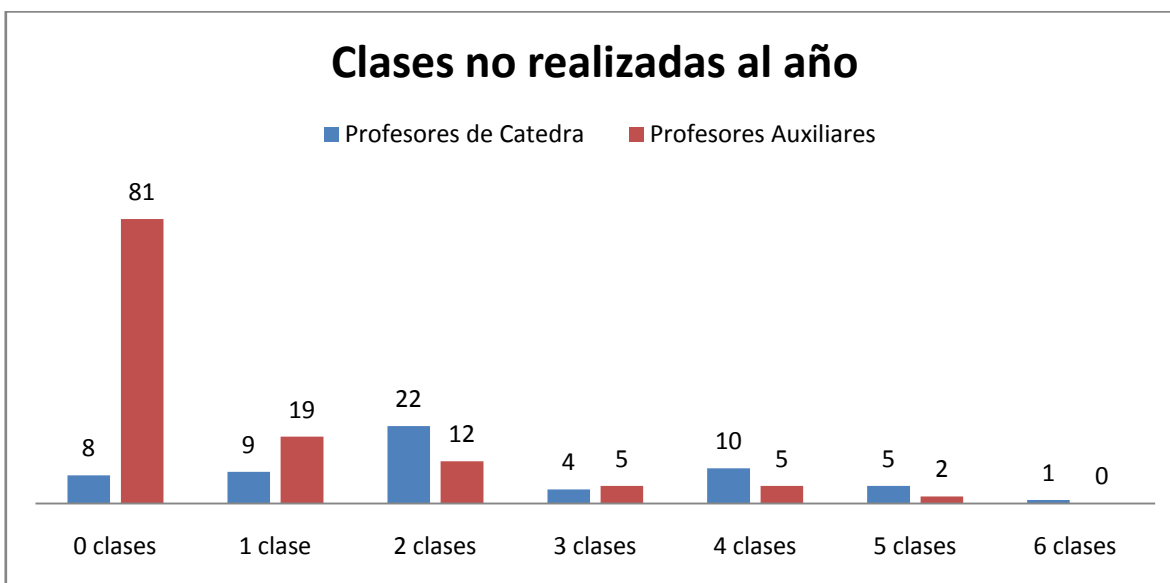


Gráfico 2 Promedio de clases no realizadas al año

Si bien los profesores auxiliares en general no pierden clases por viajes, si lo hacen los Profesores de Cátedra, de los cuales el 71% pierden 2 o más clases anuales. Esto valida la existencia de una herramienta que pueda ayudar a disminuir estas clases perdidas.

Otro de los temas tratados en la encuesta fue la disposición a participar en un AVA para realizar sus clases, obteniendo los siguientes resultados:

En primer lugar, los profesores de cátedra muestran una alta disposición a participar en un AVA, como se muestra en el Gráfico 3.

Disposición Profesores de Catédra

■ Si ■ No ■ No contesta

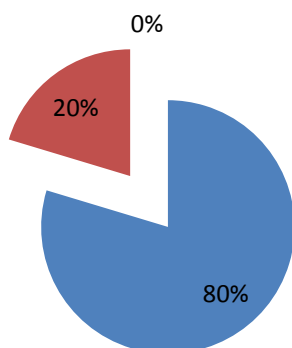


Gráfico 3 Disposición de Profesores de Cátedra

Con respecto a la disposición a participar en un AVA por parte de los Profesores Auxiliares fue menor al anterior como se muestra en el Gráfico 4.

Disposición Profesores Auxiliares

■ Si ■ No ■ No contesta

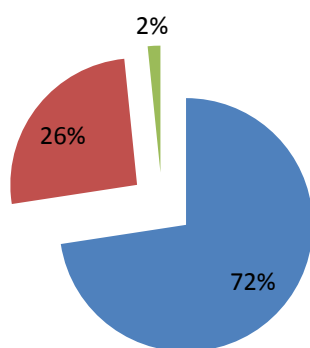


Gráfico 4 Disposición Profesores Auxiliares

Complementariamente a esta pregunta, se realizó una pregunta abierta para justificar esta respuesta, de la cual se explican algunos temas.

- Los docentes que no se encuentran dispuestos a participar argumentan que este sistema no aplica a sus metodologías de enseñanzas (por ejemplo desarrollos matemáticos, talleres prácticos específicos, etc.). Otra razón mencionada fue que las clases presenciales son insustituibles.
- La menor disposición de los Profesores Auxiliares a participar se debe principalmente al modelo de clases que se utiliza para estas actividades, por ejemplo reforzamiento de técnicas de desarrollo de ejercicios en particular.
- Las principales razones para estar dispuesto a participar en un AVA, son:
 - Es innovador
 - Es más motivador para los alumnos
 - Es posible realizar clases a distancia de forma fácil y cómoda
 - Permite realizar simulaciones de forma fácil
 - Baja asistencia a clases presenciales

1.3.1.2 Factibilidad Técnica

En cuanto a temas de factibilidad técnica de utilizar un AVA, se consultó en particular sobre el expertise y nivel de uso de los docentes del computador, además de los sistemas operativos preferidos.

El grado de uso de la tecnología en clases por parte de los docentes fue medida en 3 niveles:

- Bajo: Uso esporádico de presentaciones en powerpoint.
- Medio: Uso de presentaciones powerpoint la mayoría de las clases.
- Alto: Demostraciones varias asistidas por computador, y promover uso de ellas a los alumnos.

En este ítem los Profesores de Cátedra mostraron los resultados del Gráfico 5.

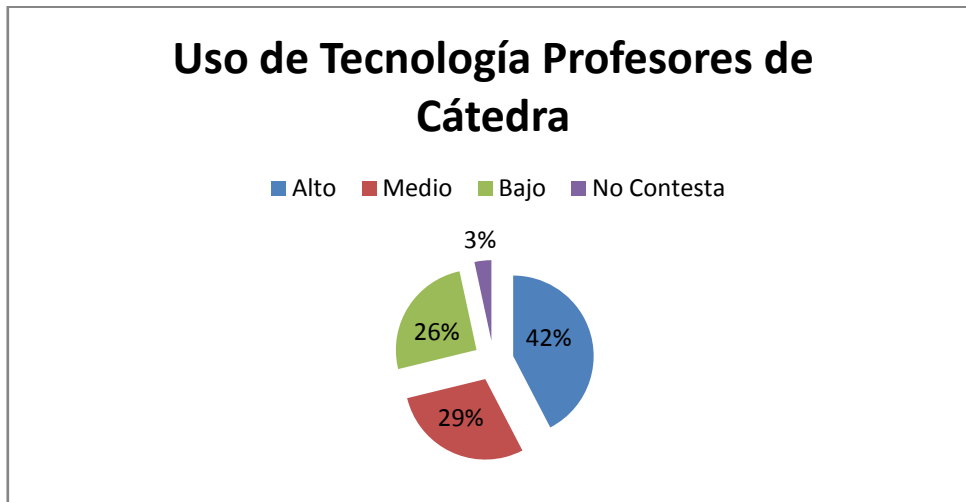


Gráfico 5 Uso de tecnología, Profesores de Cátedra

Los Profesores Auxiliares por su parte muestran los resultados del Gráfico 6.

Uso de Tecnología Profesores Auxiliares

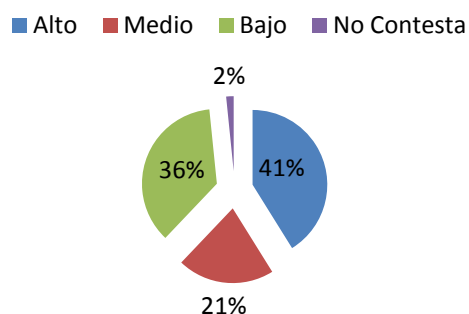


Gráfico 6 Uso de tecnología, Profesores Auxiliares

Otro tema a medir fue el expertise de los docentes en el uso del computador, definiendo los siguientes rangos:

- Bajo: El docente necesita ayuda para realizar tareas como enviar un correo.
- Medio: El docente puede realizar tareas básicas sin ayuda, como escribir un documento o hacer una planilla de cálculo.
- Avanzado: El docente no necesita ayuda para realizar tareas, y aprende rápidamente nuevas cosas.

Los Profesores de Cátedra en su mayoría se declaran como usuarios avanzados, como muestran los resultados del Gráfico 7.

Expertise Computacional Profesores de Cátedra

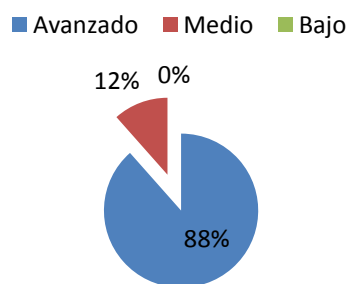


Gráfico 7 Expertise computacional, Profesores de Cátedra

Por otro lado los Profesores Auxiliares también en su mayoría se declaran usuarios avanzados, como lo muestra el Gráfico 8.



Gráfico 8 Expertise computacional, Profesores Auxiliares

Otro punto utilizado para medir la factibilidad técnica de la solución fue consultar los Sistemas Operativos más usados por el total de los docentes, entregando los resultados mostrados en el Gráfico 9.

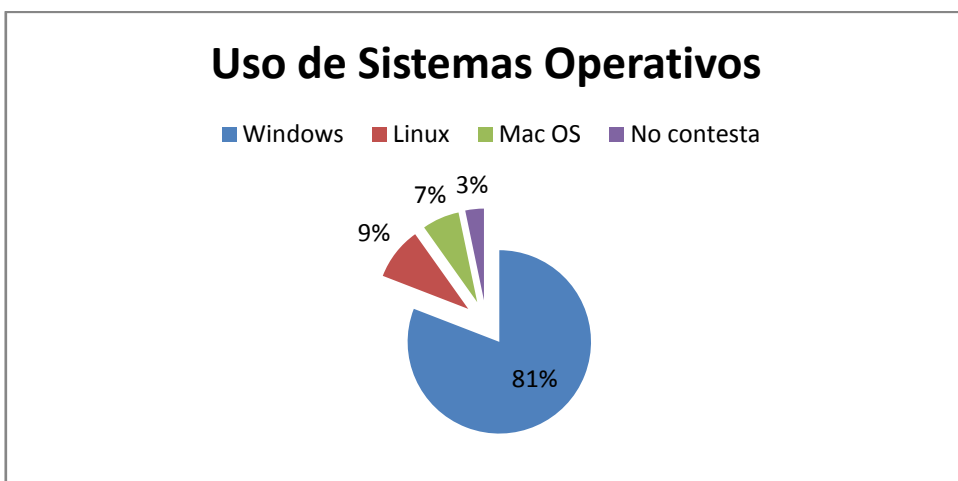


Gráfico 9 Uso de Sistemas operativos

En este punto validamos el uso de Second Life, pues existen clientes estables tanto para Windows como para Mac OS. Por otro lado, para los docentes que utilizan distribuciones Linux existe un cliente en fase experimental, con lo que no se excluiría a nadie.

1.3.1.3 Recursos Esperados

A la hora de preguntar los recursos esperados por los docentes en el AVA, bajo un grupo acotado de herramientas las elecciones fueron las mostradas en el Gráfico 10.

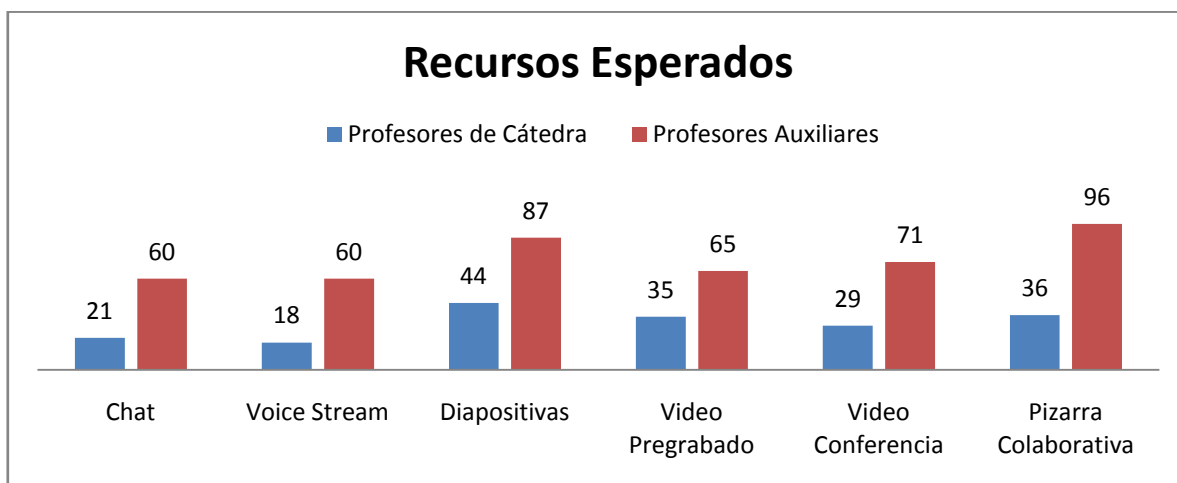


Gráfico 10 Recursos Esperados

Todas las herramientas mencionadas anteriormente existen en Second Life, y la utilización de ellas se verá a lo largo de la implementación considerando la factibilidad técnica de integración con U-Cursos.

1.3.2 Validación de Second Life

Otro estudio que se realizó fue el proceso de validación de Second Life como el AVA a utilizar. Para ello se analizaron y testearon alrededor de 20 mundos virtuales y su factibilidad para utilizarse como un AVA, analizando sus ventajas y desventajas en comparación a la alternativa seleccionada.

1.3.2.1 Primeros Descartados

A priori fueron descartados 7 mundos virtuales dadas sus características limitadas tanto en capacidad de comunicación de avatares (chat con frases pre hechas), creación de contenido limitado o nulo, y temáticas contextualizadas y restrictivas.

Estos mundos virtuales son los siguientes:

Club Penguin [11], Habbo Hotel [12], World of Warcraft [13], Nicktropolis [14], Meez [15], Zwink [16] y Roma Victor [17].

1.3.2.2 Segundos Descartados

Se revisaron 5 mundos virtuales que presentaban características más apropiadas y con un menor grado de restricciones que los revisados anteriormente. En la Tabla 2 comparativa se ven sus ventajas y desventajas con respecto a Second Life.

Tabla 2 Comparativas Second Life vs Competencias

Mundo Virtual	Ventajas con respecto a Second Life	Desventajas con respecto a Second Life
Alice [18]	<ul style="list-style-type: none">• Creación de simulaciones con facilidad	<ul style="list-style-type: none">• No es precisamente un mundo virtual sino un lenguaje para crear contenido educativo en mundos virtuales.
Twinity [19]	<ul style="list-style-type: none">• Gratuito	<ul style="list-style-type: none">• Fase Beta• Creación de Contenido Limitada
Project Wonderland [20]	<ul style="list-style-type: none">• Multiplataforma	<ul style="list-style-type: none">• Muchos errores• Poco intuitivo
Google Lively [21]	<ul style="list-style-type: none">• Corre sobre navegador• Gratuito	<ul style="list-style-type: none">• Descontinuado²• Muy pesado de ejecutar• No dispone facilidades para el aprendizaje

² <http://googleblog.blogspot.com/2008/11/lively-no-more.html>, blog oficial de Google que anuncia la discontinuación de Lively.

1.3.2.3 Los Competidores

Finalmente se encontraron los verdaderos competidores de Second Life, en términos de su utilización como AVA's. Esto dado su experimentación en enseñanza por otras entidades educacionales, herramientas especializadas y planes educativos con trayectoria.

1.3.2.3.1 Active Worlds

Este mundo virtual provee un cliente especial que apunta a la enseñanza en ambientes virtuales, entregando una gama de herramientas interesantes para dicho fin.

Active Worlds [22] contiene un cliente mucho más liviano en términos gráficos que Second Life, sin embargo actualmente sólo está disponible la versión para Microsoft Windows. Para acceder a un plan educativo en Active Worlds se debe pagar un costo anual y un costo mensual por mantención, con precios muy similares a Second Life (alrededor de US700 anuales y US150 mensuales³).

En términos de desarrollo, existen menos entidades educacionales que han preferido utilizar Active Worlds como AVA, pues dada su similitud de costos, su menor potencial y soporte en comparación a Second Life, lo han hecho perder la batalla.

1.3.2.3.2 Open Simulator

Este mundo virtual está basado en Second Life, pero es la propuesta Open Source lo que le da una importante ventaja con respecto a su competencia. Al igual que Second Life el cliente está disponible para los Sistemas Operativos más populares (Microsoft Windows, Mac OS, Linux).

Open Simulator [23] cuenta con la mayoría de las herramientas y capacidades de Second Life, y es una alternativa interesante a la hora de evaluar un AVA. Sin embargo, aún se encuentra en fase experimental y carece de algunas características fundamentales como la creación de grupos y la comunicación por voz. Esto resulta una desventaja abrumadora con respecto a Second Life, lo que finalmente nos obliga a descartarla⁴.

³ Precios referenciales con fecha Diciembre 2008.

⁴ Todas las características citadas con fecha Diciembre 2008.

1.3.2.3.3 Croquet

Croquet [24] es un proyecto open source que provee un entorno de desarrollo de mundos virtuales enfocados en su totalidad a la educación.

Croquet se promueve como la verdadera solución educativa virtual, y se presenta en la actualidad como el competidor más fuerte de Second Life, en términos de enseñanza.

Croquet funciona bajo una red p2p, lo cual lo hace escalable en términos de números de usuarios concurrentes, en contraste con el modelo cliente-servidor con que funciona Second Life.

Croquet provee un navegador 3D llamado Cobalt para interactuar con los mundos educativos creados en Croquet. Cobalt sin embargo no resulta en la actualidad una solución viable, en términos de usabilidad, como en términos de estabilidad pues aún se encuentra en fase experimental.

La principal característica que hace atractivo a Cobalt es la facilidad de interacción de documentos con texturas planas (Pdf, documentos de Word, Diapositivas, plantillas de cálculos, incluso navegadores inmersos en el mundo virtual).

1.3.2.3.3.1 Qwaq

Es un mundo virtual desarrollado en base a Croquet, con el objetivo de asistir a actividades colaborativas de oficina y educacionales a distancia.

Esta herramienta provee una interfaz intuitiva y amigable para el usuario. Permite la visualización de documentos solamente arrastrándolos dentro de una pantalla, lo cual hace muy fácil su interacción.

Qwaq [25] está disponible para Microsoft Windows y Mac OS, con versiones estables de sus clientes.

Sin embargo, el gran problema de Qwaq es su alto precio, lo que lo hace inviable económicamente. El costo de uso es de \$60US por usuario al año, y tomando en cuenta que un curso promedio tiene más de 30 alumnos, tendríamos un costo de \$1800US por curso, prácticamente 2.5 veces el costo de una isla en Second Life.

1.3.2.3.2 There

There [26] es un mundo virtual enfocado a la realización de actividades sociales, que cuenta con características como manejo de objetos, personalización de avatares e instalaciones virtuales.

Usar There es gratuito para su membresía básica incluyendo con esto la posibilidad de chatear mediante texto, personalizar el avatar y comprar objetos. Sin embargo una de las características requeridas para este trabajo, como lo es el chat de voz, esta solamente disponible para las membresías Premium, las cuales se adquieren por un pago de \$9.95US⁵ por cada avatar. La creación de objetos a pesar de estar disponible resulta algo engorrosa pues se realiza con herramientas externas a la plataforma y al finalizarlas estas deben ser enviadas a There para que evalúen si es apropiada para su uso y su comercialización. El cliente se encuentra disponible solamente para Microsoft Windows. Por estas últimas limitaciones mencionadas se descarta el uso de There para la implementación del AVA.

1.3.2.4 Second Life

Luego de un análisis de diferentes alternativas, se confirma la utilización de Second Life como el AVA adecuado para utilizar en el proyecto de memoria, pues posee grandes ventajas en el trabajo colaborativo de sus usuarios al disponer instancias para compartir información y recursos a través de diferentes medios como slides, audio y video, discusiones, presentaciones, grupos de proyectos, simulaciones de efectos físicos, capacidad de construcción, exploraciones, entre otras [27].

A continuación se detallan las características principales que posee Second Life.

⁵ Precio de Membrecía Premium en There en Agosto de 2009.

1.3.2.4.1 Herramientas y Características Generales

Second Life provee diferentes herramientas y características dentro del AVA que facilitan la interacción y el aprendizaje de sus usuarios:

- Avatares: Los usuarios pueden utilizar una identidad personalizada dentro del mundo virtual, llamada avatar, con la cual pueden interactuar y moverse en el AVA.
- Movimiento: Los avatares tienen diferentes formas de moverse a través del AVA, entre ellas caminar, volar y teletransportarse.
- Comunicación: Se provee la comunicación de los usuarios a través de chat, y una interfaz de comunicación por voz, esta comunicación puede ser tanto pública como privada.
- Objetos: Los usuarios pueden construir e interactuar con diferentes objetos en el mundo virtual, los cuales facilitan la enseñanza en términos de entregar información relevante a través de multimedios y de realizar simulaciones del mundo real.
- Scripts: Todos los objetos existentes en Second Life pueden tener un comportamiento asociado en su interacción con los avatares, los cuales son programables a través de Scripts por los usuarios en el lenguaje *Linden Scripting Language* provisto por Second Life. Estos scripts constituyen una herramienta poderosísima, sobre todo para la creación de contenido educativo.

1.3.2.4.2 Características del AVA

Second Life presenta características especiales que lo hacen un AVA adecuado para la enseñanza como las siguientes:

- Permite acceso restringido a los residentes a las regiones donde se imparten actividades de enseñanza, disminuyendo así potenciales problemas en las clases que se realicen (por ejemplo invasión de residentes que afecten el desarrollo de la clase con conductas indebidas).
- Permite controlar el número máximo de residentes permitidos en la región.
- Las regiones educacionales son privadas, esto quiere decir que están separadas de la MainLand (Isla Principal), lo que las hace exclusivas para los residentes autorizados.
- Permiten un diseño personalizado de la región, tanto en texturas como en formas.
- Cada región privada corre sobre 1 servidor dedicado, lo que asegura un desempeño adecuado de la plataforma.

1.3.2.4.3 Factores Económicos

Las Instituciones educacionales pueden acceder a un importante descuento por el arriendo de regiones en Second Life. Los costos anuales ascienden a \$700US por una región de 65,536m², además de un impuesto de mantención mensual de \$147,50US⁶. Si bien el costo es importante, no es excesivo pues dentro de esta región pueden interactuar todos los usuarios que sean necesarios, sin restricciones.

⁶ <http://secondlife.com/land/privatepricing.php>

2 Marco Conceptual

2.1 Conceptos Generales

ADI: Área de Infotecnologías de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile.

Ambiente Virtual de Aprendizaje (AVA): es un sistema de software diseñado para facilitar a profesores la gestión de cursos virtuales para sus estudiantes, especialmente ayudándolos en la administración y desarrollo del curso. El sistema puede seguir a menudo el progreso de los principiantes, puede ser controlado por los profesores y los mismos estudiantes. Originalmente diseñados para el desarrollo de cursos a distancia, vienen siendo utilizados como suplementos para cursos presenciales⁷.

API: Interfaz de programación de aplicaciones (del inglés Application Programming Interface) es un conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizada por otro software como una capa de abstracción.

Educación Ubicua: Se desprende del término Computación ubicua⁸ (se le llama también *pervasive computing*) que aplicada a la educación intenta lograr que la tecnología sea “invisible” para el usuario. La meta es desarrollar entornos transparentes para que de esa manera los usuarios puedan beneficiarse de la tecnología sin “darse cuenta” que la están usando.

Lag: Proveniente del inglés lag behind (rezagarse), es el retraso producido en una comunicación que dificulta el desarrollo normal de la misma, provocando desorientación o incomodidad a sus usuarios. Las principales causas de lag en las redes informáticas son: Mal desempeño de la red, Insuficiente poder de procesamiento del servidor, Insuficiente poder de procesamiento del cliente.

Open Source: Es el termino con el que se conoce al software distribuido y desarrollado libremente. Se refiere también a los programas que se ofrecen con total libertad de modificación, uso y distribución bajo la regla implícita de no modificar dichas libertades hacia el futuro⁹.

Procesamiento Batch: Ejecución de una serie de programas o rutinas de tareas en el computador, sin la interacción humana.

⁷ http://es.wikipedia.org/wiki/Ambiente_Educativo_Virtual

⁸ <http://www.vidadigital.net/blog/2006/05/12/dvd-gratuito-sobre-computacion-ubicua/>

⁹ http://es.wikipedia.org/wiki/C%C3%B3digo_abierto

2.2 Conceptos de Second Life

Avatar: Apariencia visual de los usuarios dentro de Second Life (o de cualquier AVA), el cual puede ser un personaje humano, animal o mitológico.

Linden Dólar (L\$): Es la moneda de intercambio de todas las transacciones que se realizan en Second Life. Este puede ser transformado a diferentes divisas (USD, EUR, GBP, JPY) en el mercado Lindex¹⁰ y otros.

Linden Scripting Language (LSL): Es el lenguaje en que están escritos todos los scripts dentro de Second Life. Su estructura está basada en Java y C. Un script en Second Life es un conjunto de instrucciones que pueden ser puestas dentro de cualquier objeto en el mundo, pero no dentro de los avatares. El lenguaje puede ser escrito a través de un editor-compilador. LSL es interpretado y ejecutado en los servidores de Second Life y no en el cliente, aunque el editor sea parte de este. LSL es especial dado su énfasis en los estados y eventos, esto pues en la vida real la mayoría de los objetos presentan estados (por ejemplo, una puerta puede estar abierta o cerrada). Mínimamente un script puede tener un estado, llamado estado por defecto¹¹.

MainLand: Región Principal de Second Life donde interactúan por defecto la mayoría de los residentes.

MegaPrim: Es un prim con al menos uno de sus lados más largo que 10 metros. No pueden ser conseguidos por las herramientas de Second Life, estos deben ser transferidos por otro avatar que los posea.

Notecard: Ítem del inventario que contiene texto en su interior por defecto y que puede ser editado por los avatares.

Prim: Es un objeto geométrico tridimensional e individual, los cuales pueden crearse a partir de las siguientes formas: Caja, Cilindro, Prisma, Esfera, Toro, Tubo y Anillo, y pueden ser alterados con las herramientas de edición disponibles. Cada prim es representado por un conjunto de parámetros, incluyendo posición, escala, rotación, forma, cavidades, etc. Los prims pueden ser concatenados para crear estructuras espaciales más complejas.

Residentes: Usuarios que residen o participan habitualmente en el mundo virtual de Second Life.

Teletransportación: Metáfora que se utiliza cuando una persona ingresa al mundo virtual de Second Life desde una URL o se traslada desde una ubicación a otra dentro de Second Life.

¹⁰ <http://secondlife.com/statistics/economy-market.php>

¹¹ http://wiki.secondlife.com/wiki/Help:Getting_started_with_LSL#What_is_LSL.3F

2.3 Tecnologías Asociadas

Apache HTTP Server Project: Es un servidor HTTP de código abierto, seguro, eficiente y extensible para sistemas UNIX y derivados, además para plataformas de Microsoft Windows. En la actualidad es una de las soluciones más empleadas para servir el alojamiento de páginas Web. Permite una fácil integración con diferentes lenguajes de programación como PHP, para lograr contenido dinámico.

Entorno de Desarrollo: El ambiente de desarrollo fue provisto por el Área de Infotecnologías (ADI), de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile. La programación y estructura del sistema se llevó a cabo siguiendo los estándares definidos por dicha entidad. El sistema funciona con un servidor Linux con Apache HTTP Server, PHP como principal lenguaje de Programación y MySQL como motor de base datos. Además para llevar un control de versiones se utilizó Subversion (SVN).

Hypertext Processor (PHP): Es un lenguaje de programación basado en scripts, el cual fue diseñado específicamente para ser usado en el desarrollo de páginas Web. Existe una gran cantidad de funciones y soporte en línea de este lenguaje.

InfoNúcleo: Sistema centralizado de configuraciones de módulos que posee ADI.

MySQL: Es un motor de base de datos de código abierto. Su sintaxis se basa en el lenguaje estándar SQL (Simple Query Language), diseñado especialmente para realizar consultas sobre base de datos. MySQL cuenta con un mecanismo de almacenamiento basado en InnoDB que permite la realización de transacciones de forma segura, además cumple con el estándar ACID, ofreciendo funciones de Commit, Rollback y recuperación en caso de fallas.

PEAR (PHP Extensión and Application Repository): Es un entorno de desarrollo y sistema de distribución para componentes de código PHP¹².

Red Peer to Peer (P2P): Red descentralizada que no tiene clientes ni servidores fijos, sino que tiene una serie de nodos que se comportan simultáneamente como clientes y servidores de los demás nodos de la red. Cada nodo puede iniciar, detener o completar una transacción compatible¹³.

Subversion (SVN): Es un programa de código abierto cuya finalidad es tener control de las versiones de un programa en su proceso de desarrollo, facilitando el trabajo en equipo de forma paralela, con la principal característica de ofrecer en cualquier momento volver a una versión previa de un programa.

¹² <http://pear.php.net/>

¹³ <http://www.alegsa.com.ar/Dic/p2p.php>

3 Metodología

3.1 Análisis de Situación Actual

En esta etapa se identificó el problema principal a resolver, y se estudiaron las características y funcionalidades de Second Life en relación al proceso de aprendizaje en temas como:

- Funcionamiento general de SL.
- Funcionamiento de SL como herramienta de aprendizaje.
- Herramientas multimedia de apoyo a la docencia en SL.
- Mejores Prácticas y Experiencias de otras Universidades en SL.
- Interacción de SL con plataformas Web.

Por otro lado también se investigó acerca del ambiente de desarrollo de ADI para la integración de nuevos módulos, estándares de programación y buenas prácticas para el proceso de construcción del módulo de administración.

3.2 Requerimientos del Sistema

En esta etapa se realizó una definición general de los principales desafíos a resolver para la integración de Second Life con U-Cursos, lo que incluyó:

1. Levantamiento de Requisitos del módulo de configuraciones de U-Cursos.
2. Levantamiento de Requisitos de las herramientas en Second Life.
3. Definición del alcance del Proyecto.

3.3 Metodología de Desarrollo

3.3.1 Aprendizaje de Interacción U-Cursos y Second Life

Se realizó un estudio de cómo utilizar las herramientas que provee Second Life para interactuar con plataformas Web y programar las características necesarias para dicha comunicación.

3.3.2 Aprendizaje Interacción, Creación de Objetos y Scripts en Second Life

Se revisó y estudió una amplia gama de tutoriales online [28][29][30][31][32], acerca de cómo interactuar en el mundo virtual, cómo construir objetos en base a primitivas y creación de scripts para entregarles características interactivas a estos.

3.4 Diseño del Sistema

En este punto se diseñó la lógica de las distintas componentes tanto interfaces Humano U-Cursos, como las interfaces U-Cursos Second Life que permiten gestionar y administrar los recursos disponibles del proyecto. Además se diseñó la parte lógica y física de los objetos virtuales de Second Life con sus respectivos scripts de comportamiento que permiten tener la experiencia interactiva de una clase.

3.5 Implementación del Sistema

En esta etapa se utilizó un modelo de desarrollo iterativo e incremental, con esto se lograron rápidos prototipos funcionales de las partes del sistema, que en cada iteración eran mejorados tanto en funcionalidades como en performance. Cada iteración se puede resumir con los siguientes pasos:

1. Etapa de inicialización: Creación de un producto con el cual el usuario puede interactuar, el cual provee una solución simple al problema o requisito identificado.
2. Etapa de iteración: En esta etapa se rediseñaron e implementaron nuevas funcionalidades o se mejoraron las ya existentes de las diferentes partes del sistema, dados los requisitos del sistema o el feedback entregado por la contraparte (en este caso el cliente ADI) en las diferentes reuniones de revisión.

4 Análisis Situación Actual

4.1 Descripción del Problema

En la actualidad, el proceso de impartir una clase presencial en el contexto universitario puede sufrir ciertos inconvenientes para poder realizarse como lo es la imposibilidad física de asistir a una sala, por diferentes motivos. Sumado a lo anterior, existen ocasiones en que las clases pueden ser complementadas por profesores en el extranjero, o de plano realizar clases compartidas con entidades educacionales ubicadas en otra localidad. Dadas estas posibilidades, se han probado diferentes formas de suplir las dificultades anteriormente mencionadas a través de tecnologías como videoconferencias, materiales multimedia en línea, listas de correos, o cualquier forma de E-Learning existente.

Dado este problema, surge la necesidad del ADI por entregar una herramienta complementaria para apoyar este factor, el cual debe integrarse fácilmente con su plataforma educacional U-Cursos, y entregar a los docentes y alumnos una alternativa para esta dificultad, encontrando en un Ambiente Virtual de Aprendizaje (AVA) una buena solución.

Un AVA resulta un recurso ideal como herramienta complementaria a la hora de de realizar clases, dadas las siguientes características:

- Provee una interfaz de usuario interactiva tridimensional, novedosa y motivante a cada uno de los actores participantes.
- Ahorro de recursos en comparación a soluciones parecidas como una videoconferencia, que tienen un alto costo asociado a la factibilidad técnica y sincronización de componentes.
- Posibilidad de realizar simulaciones de eventos y objetos existentes en el mundo real, contribuyendo a la comprensión práctica de estos.

4.2 Second Life en la Educación

El mundo virtual de Second Life ya ha sido explorado y explotado en el campo de la educación por muchas instituciones, esto pues cuenta con características como:

- Acceso gratuito a la mayoría de las características.
- Ofrece diferentes elementos multimedia (video, música, chat escrito y por voz, objetos interactivos y programables, etc.) con un amplio soporte y documentación en línea [33][34][35].
- Permite la creación de objetos y ambientes personalizados, a través de las diferentes herramientas de fácil uso.
- Entrega diferentes herramientas de administración de permisos a los dueños de terrenos para controlar la conducta de los avatares que visitan sus dependencias.

Todas estas características y funcionalidades respaldan la creación de un aula virtual para entregar a la comunidad universitaria una nueva herramienta para apoyar los procesos de enseñanza-aprendizaje.

5 Requerimientos del Sistema

5.1 Problemática

Se requiere construir un sistema capaz de organizar y facilitar la realización de una clase en el mundo virtual de Second Life con un conjunto de herramientas básicas que provean al docente las herramientas necesarias para la exposición de los contenidos. El sistema además debe contar con un módulo dentro de la plataforma educativa U-Cursos, donde los docentes puedan preparar y organizar su clase, y los alumnos puedan enterarse de los detalles de ésta, además de indicarles las instrucciones necesarias para asistir a la misma.

Esta herramienta por ahora sólo estará disponible para la Facultad de Ciencias Físicas y Matemáticas (FCFM) de la Universidad de Chile, y será un módulo disponible para todos los cursos que se imparten, entregándola como un complemento opcional para los docentes que lo requieran.

5.2 Requerimientos Funcionales

5.2.1 Requerimientos en U-Cursos

1. Creación de un nuevo módulo para la plataforma web educativa U-Cursos, la cual debe estar asociada a todos los cursos que se imparten en la FCFM. Este módulo recibirá el nombre de *Clases Virtuales*.
2. El módulo *Clases Virtuales* debe proveer al cuerpo docente de un curso (Profesor de Cátedra, Profesor Auxiliar) la posibilidad de preparar una clase virtual, a la cual se le deben asignar los siguientes datos:
 - Fecha
 - Hora de Inicio y de Fin
 - Reserva de Sala
 - Adjuntar Diapositivas
 - Definir privacidad de la clase según las siguientes alternativas:
 - i. Clase Abierta: Cualquier persona (dentro y fuera del contexto universitario) puede entrar y participar en la clase.
 - ii. Clase Cerrada: Solo un conjunto acotado de alumnos puede participar en la clase, y el docente debe tener la capacidad de seleccionarlos a partir de la lista oficial del curso.
3. El módulo de *Clases Virtuales* debe tener las siguientes opciones para las asignaciones ya realizadas por los docentes:

- Edición: Las clases virtuales ingresadas podrán ser modificadas antes de que la fecha de inicio de la misma expire.
 - Borrado: Las clases virtuales ingresadas podrán ser eliminadas.
 - Publicación: Las clases virtuales creadas, pueden ser Publicadas/Despublicadas por los docentes, lo cual permite la visualización de la misma por parte de los alumnos en las nuevas publicaciones del curso.
4. Los alumnos sólo podrán acceder a la información de las clases virtuales debidamente publicadas por los docentes, pudiendo acceder a la calendarización de la misma, además de entregarle las instrucciones necesarias para acceder a ella.
 5. Todas las interfaces de usuario deben conservar el patrón de diseño de la plataforma U-Cursos, además de cumplir las normas de programación entregadas por ADI [36].

5.2.2 Requerimientos en Second Life

1. Se debe proveer de un *Aula Virtual* que simule en el mayor grado posible una sala de clase real, que sea capaz de restringir el acceso a los avatares no invitados a la clase en el caso de seleccionar una clase privada.
2. Se debe proveer un objeto o algún elemento capaz de mostrar la presentación entregada por el profesor a través de U-Cursos.

5.2.3 Requerimientos Globales

1. Se debe crear una interfaz de comunicación capaz de integrar la plataforma de U-Cursos y el mundo virtual de Second Life, con la cual se pueda coordinar el proceso de preparación e impartición de una clase virtual.

5.3 Requerimientos No Funcionales

- El sistema debe ser eficiente en el uso de recursos de los servidores involucrados. En la plataforma de U-Cursos, se debe tomar en cuenta que debe integrarse sin realizar una demanda importante de recursos y en el mundo virtual de Second Life, se debe disponer de los recursos con la mayor simpleza posible.
- El sistema deberá ser seguro en todas sus capas, contando con un control de acceso diferenciado a los distintos usuarios del sistema.
- El sistema debe ser amigable. Las interfaces deben ser intuitivas para sus usuarios, sin romper el esquema de las plataformas involucradas y privilegiando la simpleza de ellas.
- El sistema debe ser escalable, es decir, debe permitir la extensión futura de todas sus componentes de forma fácil y sin un gran impacto.
- El sistema debe ser Robusto. Toda la información que sea procesada en el sistema debe ser consistente.

6 Diseño de la Solución

6.1 Descripción de la Solución

El diseño del sistema completo se subdivide en 3 componentes principales:

1. Módulo web *Clases Virtuales*, el cual estará inmerso en la plataforma U-Cursos, donde se prepara y organiza una clase.
2. Objetos Interactivos que componen el *Aula Virtual* dentro de Second Life, donde los usuarios podrán participar en una clase.
3. Interfaces de comunicación entre el módulo *Clases Virtuales* y los objetos interactivos de Second Life, que realizan la integración de ambas plataformas, compartiendo información relevante para la realización de las clases.

6.1.1 Características de los Usuarios

Los usuarios del sistema son diferenciados en 2 tipos según los privilegios que posean en el sistema de U-Cursos, con las características mostradas en la Tabla 3.

Tabla 3 Usuarios del Sistema

Tipo de Usuario	Descripción	Funcionalidades Disponibles
Docente	En esta categoría entran los Profesores de Cátedra y Profesores Auxiliares de todos los cursos que se encuentren disponibles en U-Cursos de la FCFM	<ul style="list-style-type: none">• Pueden agregar, ver, editar, publicar y eliminar clases virtuales en U-Cursos.• Son los encargados de Dictar la Clase Virtual en Second Life, y tendrán las facultades para:<ul style="list-style-type: none">○ Manejar la presentación○ Expulsar avatares del aula virtual
Alumno	Son todos los alumnos de la FCFM que tengan cursos inscritos en U-Cursos.	<ul style="list-style-type: none">• Pueden visitar los datos de las clases virtuales debidamente publicadas (link de teletransporte, presentación, tipo de clase, etc.)• En Second Life pueden participar en las clases como espectadores, al igual que en las aulas de clases normales, pudiendo realizar preguntas o dar opiniones bajo un ambiente de orden y respeto.

6.1.2 Visión General

Los objetivos del sistema se resumen en 2 procesos, que se pueden ver a continuación:

1. Preparación de la Clase: Realizada por el usuario docente a través de U-Cursos, y que es descrito en el diagrama de flujo de la Figura 1.

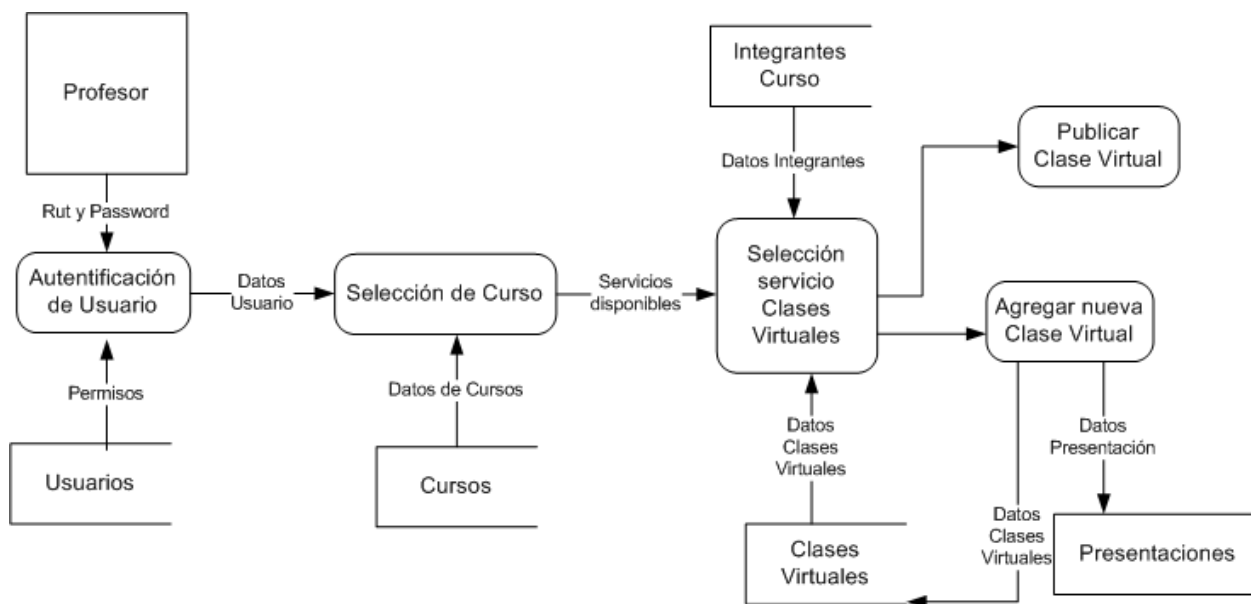


Figura 1 Preparación Clase Virtual

2. Ingreso Clase Virtual: Tanto el docente como el alumno deben realizar el mismo proceso para asistir al aula virtual y participar en la clase, descrito en el diagrama de flujo de la Figura 2.

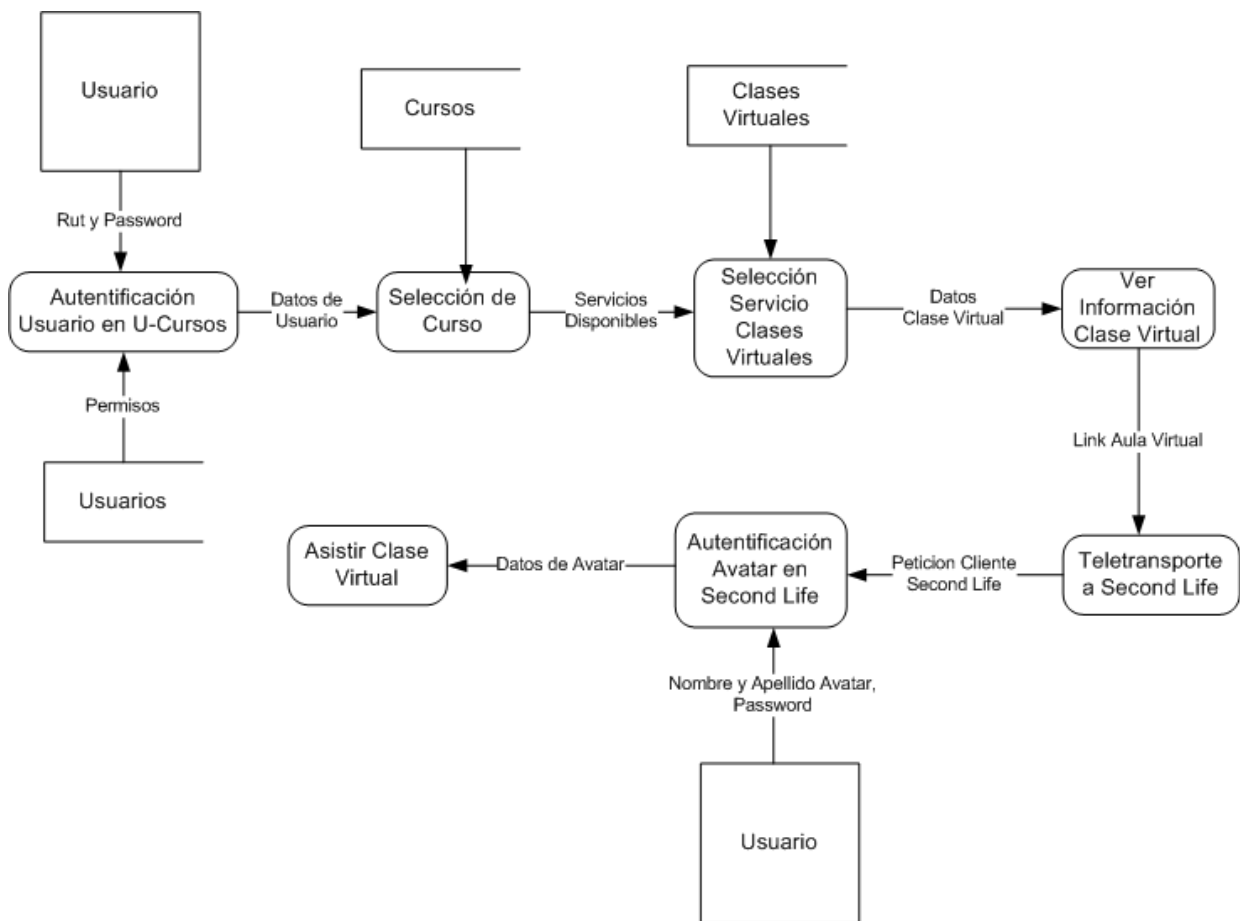


Figura 2 Ingreso Clase Virtual

6.2 Diseño Arquitectónico

6.2.1 Arquitectura Física

La arquitectura del sistema, en base a las 3 componentes principales, se divide en las siguientes partes:

1. Arquitectura Física parte U-Cursos: Representa la interacción de los usuarios del sistema con el módulo web *Clases Virtuales* presente en la plataforma U-Cursos.
2. Arquitectura Física parte Second Life: Representa la interacción de los usuarios del sistema con el Aula Virtual y todos los objetos creados en el mundo de Second Life.
3. Arquitectura Física parte U-Cursos Second Life: Representa la interacción que ambas plataformas tienen para intercambiar los datos de configuración de las clases virtuales preparadas en el sistema.

6.2.1.1 Arquitectura Física parte U-Cursos

La plataforma U-Cursos presenta un modelo de 3 capas tradicional en los sistemas Web, donde los usuarios se deben conectar mediante cualquier navegador de internet a él. Este muestra una interfaz gráfica que carga todas las aplicaciones alojadas en el servidor Web, el cual contiene la lógica de negocios desarrollada con el lenguaje de programación PHP, que a su vez interactúa con la información almacenada en el motor de base de datos MySQL, el cual retorna al usuario con los resultados requeridos, como se muestra en el esquema de la Figura 3.

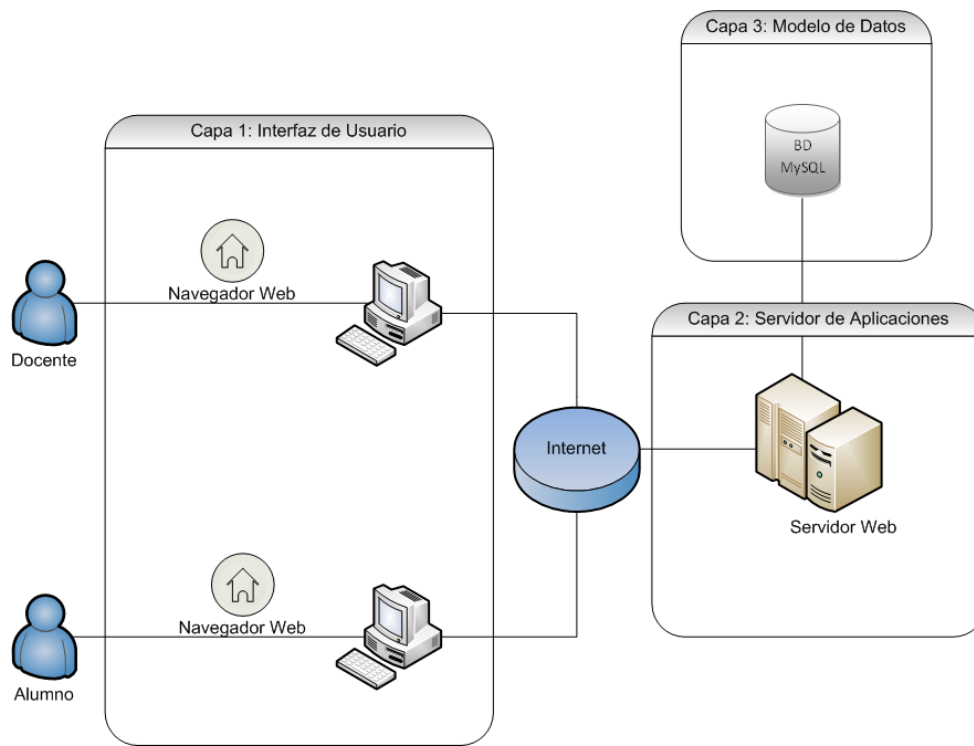


Figura 3 Arquitectura Física parte U-Cursos

6.2.1.2 Arquitectura Física Parte Second Life

Second Life funciona bajo el modelo Cliente-Servidor. El cliente instalado en el computador del usuario envía el contenido generado por el usuario (acciones, objetos, scripts) y representa gráficamente los objetos que el servidor le proporciona. El servidor, por su parte, es el encargado de mantener la consistencia del mundo virtual, hacer seguimiento de la posición de los objetos, procesar físicas y transparencias y decidir cuál es la información que se envía a los clientes. El servidor principal utiliza 3 tipos de máquinas:

- **Servidores de datos:** maneja las conexiones con la base de datos central, base de datos de registros, inventarios, y búsqueda.
- **Servidores de simulación del mundo virtual:** realiza los cálculos de los objetos y terrenos y transmite los datos al cliente.
- **Servidores de Usuarios:** encargado de la autenticación y mensajería instantánea.
- **Servidor del Espacio:** maneja el enrutamiento de mensajes basados en la grilla. Además el servidor de simulación habla al servidor del espacio para registrar y encontrar a los vecinos.

Un esquema de la arquitectura se puede apreciar en la Figura 4.

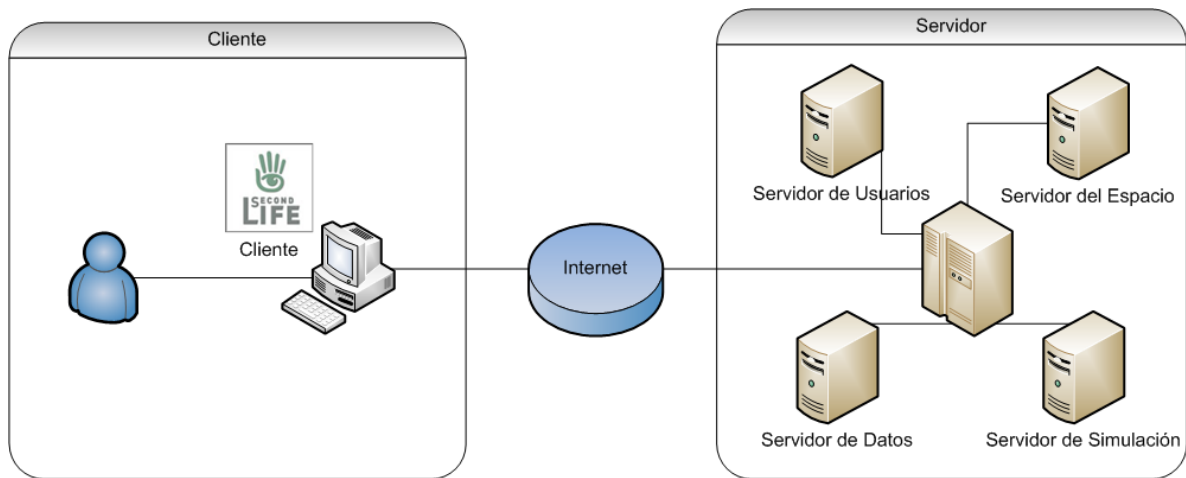


Figura 4 Arquitectura Física parte Second Life

6.2.1.3 Arquitectura Física Parte U-Cursos Second Life

La comunicación entre ambas plataformas se realizará a través de una conexión entre los servidores, donde básicamente los objetos existentes en Second Life realizarán requerimientos de datos a los servidores de U-Cursos a través de internet. Un esquema de la interacción de las plataformas es el mostrado en la Figura 5.

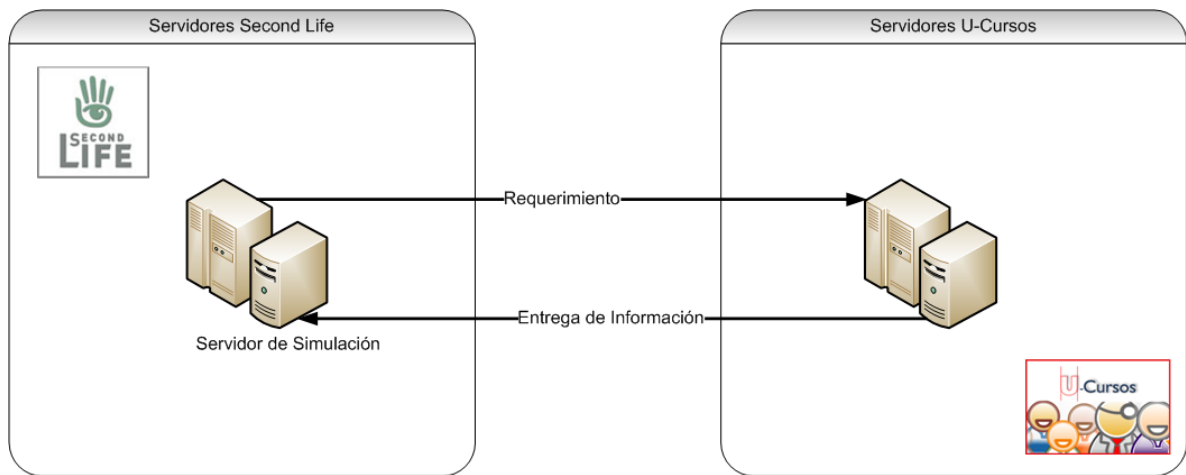


Figura 5 Arquitectura Física parte Second Life U-Cursos

6.2.2 Arquitectura Lógica

6.2.2.1 Arquitectura Lógica parte U-Cursos

La arquitectura de esta componente del sistema está basada en el patrón de software MVC (Modelo Vista Controlador). El MVC permite separar los datos de la aplicación (Modelo, en adelante: “**Model**”), la interfaz de usuario (Vista, en adelante “**View**”), y la lógica de control (Controlador, en adelante “**Controller**”) en tres componentes diferentes y separadas explícitamente.

El diagrama de la Figura 6, muestra las relaciones entre las tres componentes. Las líneas sólidas representan una asociación directa y las punteadas una indirecta.

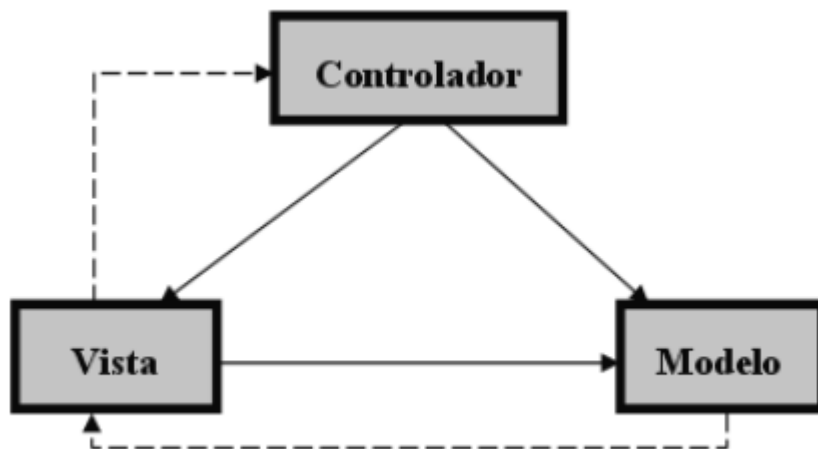


Figura 6 Patrón MVC

6.2.2.1.1 Capa de Vista

La capa de vista es la interfaz que se muestra directamente a los usuarios mediante el navegador web. Las páginas se muestran de forma dinámica, y para el ingreso de información se proveerá de formularios de datos adecuados, que minimicen la probabilidad de error al ingresar los datos requeridos.

6.2.2.1.2 Capa de Control

La capa de Control es la encargada de recibir la interacción de los usuarios del sistema con la capa de vista, maneja la lógica de negocios y entrega las respuestas a las peticiones de usuario. Las componentes identificadas del sistema son las mostradas en la Figura 7.

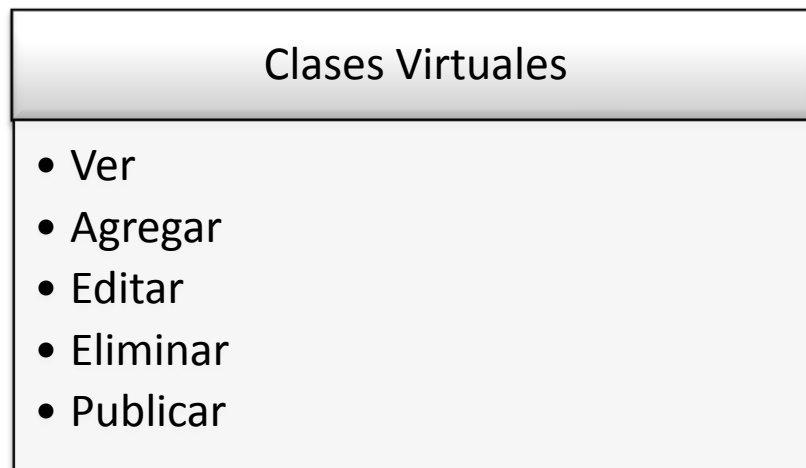


Figura 7 Mantenedor Clases Virtuales

El mantenedor de clases virtuales proveerá todas las funcionalidades asociadas para la preparación y organización de las clases que se impartan en Second Life. Este mantenedor será usado por los diferentes usuarios del sistema, restringidos según los permisos asociados a sus perfiles.

6.2.2.1.3 Capa de Modelo

Esta capa corresponde al modelo de datos asociado a la base de datos, la cual almacena toda la información obtenida desde el proceso de negocio. El modelo de datos será explicado en el siguiente capítulo.

6.2.2.2 Arquitectura Lógica parte Second Life

Los usuarios del sistema a través del cliente de Second Life encontrarán un conjunto de objetos tridimensionales virtuales creados a semejanza de la realidad, con los cuales podrán interactuar.

La interacción de los objetos será programada a través del lenguaje *Linden Scripting Language* (LSL) que provee Second Life, y que usa una sintaxis muy similar a los lenguajes de C y Java, pero con la diferencia que es un lenguaje con manejo de estados y eventos, lo que permite modelar comportamientos de objetos de la vida real.

Un objeto puede contener una cantidad variable de scripts corriendo simultáneamente y estos son programados o llamados en el cliente pero ejecutados en el servidor de Second Life.

Por lo general los scripts responden a las acciones que realizan los avatares, como por ejemplo hacer *click* sobre los objetos o por intercambio de mensajes con ellos. Un ejemplo simple de script es el mostrado en Figura 8.

```
default
{
    touch_start (integer total_number)
    {
        llSay (0, "Hello World");
    }
}
```

Figura 8 Ejemplo de script Hello World en Linden Scripting Language

El código anterior posee solamente un estado, el estado *default*, el cual posee a su vez sólo un evento en su interior, en el cual los avatares que hagan *click* en el objeto que contenga este script desencadenarán el evento *touch_start()*, que enviará el mensaje "Hello World".

Cabe destacar que el lenguaje de Linden Script cuenta con una amplia gama de funciones en su API, la cual se encuentra documentada y con una gran variedad de ejemplos en la wiki oficial de Second Life¹⁴.

6.2.2.3 Arquitectura Lógica parte Second Life U-Cursos

La conversación de los servidores de ambas plataformas se lleva de la siguiente forma.

6.2.2.3.1 Interfaz de Comunicación Second Life

Los objetos creados en Second Life que requieran información de U-Cursos lo harán a través del uso de scripts, como se explicó anteriormente, específicamente utilizando la función *llHTTPRequest()*. Un ejemplo de ella se ve en la siguiente figura.

```
key http_request_id;
default
{
    state entry()
    {
        http_request_id = llHTTPRequest("url", [], "");
    }

    http_response(key request_id, integer status, list metadata, string body)
    {
        if (request_id == http_request_id)
        {
            llSetText(body, <0,0,1>, 1);
        }
    }
}
```

Figura 9 Ejemplo Request Scripting Language

¹⁴ http://wiki.secondlife.com/wiki/LSL_Portal/

El código anterior contiene solamente un estado, *default*, y en él 2 eventos. El primero es el estado de entrada, *state_entry()*, el cual se ejecuta cuando el objeto que lo contiene es sacado del inventario o después de reiniciar el script, éste contiene una petición a un URL particular con la función *//HTTPRequest()*. La respuesta de la petición es manejada en el evento de respuesta, *http_response()*, el cual comprueba que la petición se haya realizada con éxito y los datos pedidos son enviados mediante formato HTML y convertidos en una cadena de texto almacenada en el parámetro *body*, el cual es mostrado sobre el objeto con la función *//SetText()*.

Es así como se realizan todas las peticiones hacia los servidores de U-Cursos, realizando el requerimiento y posteriormente los datos recibidos son manejados según la función del objeto que los requiera.

6.2.2.3.2 Interfaz de Comunicación U-Cursos

Por su parte en U-Cursos existirán un grupo de scripts, de procesamiento batch, que al ser requeridos por los objetos de Second Life, realizarán un conjunto de tareas (consultas SQL, procesamiento de texto, decisión de que información será enviada, etc.), la cual será entregada de forma conveniente para que los objetos de Second Life puedan procesarla y realizar las acciones pertinentes con ella.

6.3 Diseño Detallado

6.3.1 Clases Virtuales U-Cursos

6.3.1.1 Definición Clase Virtual

Una clase virtual será definida como un objeto conceptual que tiene las características mostradas en Figura 10 y en la Tabla 4.



Figura 10 Definición Clase Virtual

Tabla 4 Características Clase Virtual

Atributo	Características
Presentación	Es la diapositiva asociada a la clase, por restricciones de Second Life esta debe ser una secuencia de imágenes.
Sala Virtual	Es el espacio en el mundo virtual donde se realiza la clase.
Horario	Calendarización de la clase.
Permisos de Alumnos	Lista de Alumnos permitidos en la clase en el caso de una clase privada.
Profesor	El profesor que dicta la clase.
Título	Nombre de la clase.

6.3.1.2 Módulo Clases Virtuales

Este nuevo módulo será parte de los servicios que están disponibles en los cursos dictados en la FCFM.

El objetivo principal de este será entregar las herramientas necesarias para preparar y coordinar una nueva clase virtual a través de la plataforma de Second Life.

Las funcionalidades de este servicio, diferenciado por usuario, se ven en el diagrama UML de la Figura 11.

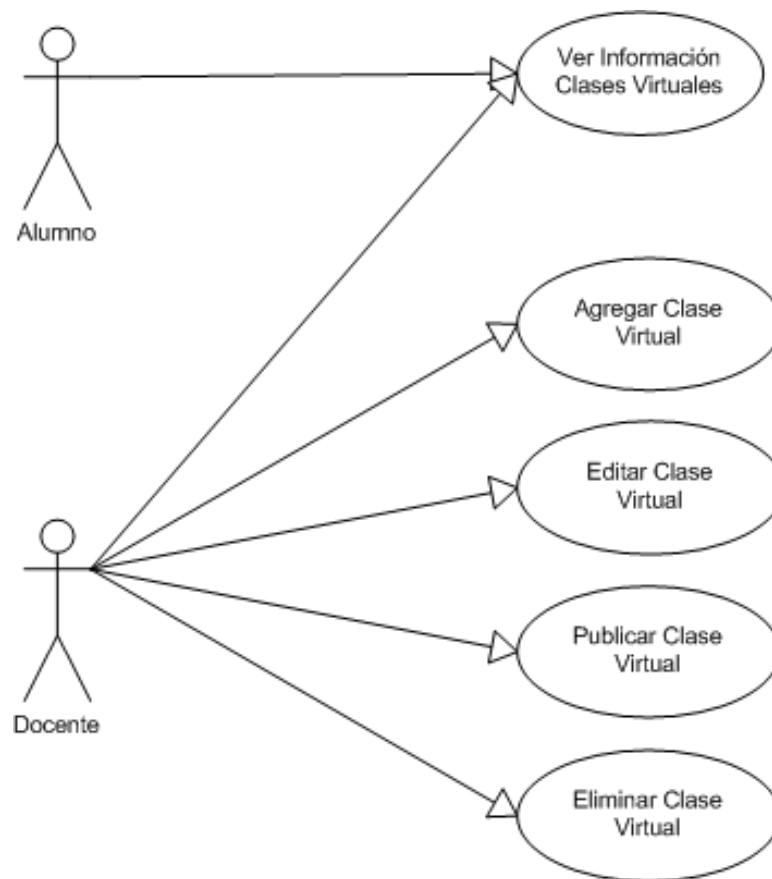


Figura 11 Diagrama UML Clases Virtuales

6.3.1.2.1 Ver Información Clases Virtuales

Esta funcionalidad está disponible para todos los usuarios del sistema, y permite ver la información asociada a la clase virtual. Tiene las características mostradas en la Tabla 5.

Tabla 5 Información Clases Virtuales

Información Clases Virtuales		
Tipo de Usuario	Restricciones	Información disponible
Alumno	Ver sólo publicadas	<ul style="list-style-type: none">• Profesor• Permisos de Alumnos• Presentación• Fecha• Horario• Descripción• Sala y Link de Teletransporte
Docente	Ver publicadas/no publicadas	<ul style="list-style-type: none">• Profesor• Permisos de Alumnos• Presentación• Fecha• Horario• Descripción• Sala y Link de Teletransporte

Cabe destacar que uno de los requisitos del módulo de Clases Virtuales es que todos sus actores deben tener en sus datos privados de U-Cursos asociado el nombre de avatar personal de Second Life, como se muestra en la Figura 12. Esto pues es la única forma de entregar los permisos de forma correcta a la sala virtual y poder restringir el acceso a otros usuarios.

Figura 12 Datos de Avatar Personal en Second Life

6.3.1.2.2 Agregar Clase Virtual

Esta funcionalidad permite agregar nuevas clases virtuales al sistema, por parte de los usuarios docentes. Las nuevas clases agregadas solo pueden ser vistas por los usuarios docentes, y para hacerlas visibles para los usuarios alumnos deben ser previamente publicadas. Los datos que debe proporcionar el usuario son los mostrados en la Tabla 6.

Tabla 6 Datos Clase Virtual

Atributo	Detalle
Título	Nombre descriptivo de la clase virtual
Fecha	Fecha de realización de la clase virtual
Horario	Horario de clase en que se realiza la clase virtual, sigue el formato de la FCFM ¹⁵
Presentación	Archivos de imagen que componen una presentación
Sala Virtual	Selección de Sala en el Mundo Virtual (en principio es una, pero puede ser extensible en el tiempo)
Avatar	Avatar utilizado por el profesor para realizar la clase
Descripción	Reseña del contenido de la clase
Privacidad	Tipo de clase y restricciones de acceso a alumnos que tendrá. La clase abierta permite a cualquier alumno ingresar y la clase privada permite ingresar sólo a los alumnos seleccionados asistir

¹⁵ Ver Anexo 10.1

6.3.1.2.3 Editar Clase Virtual

Esta funcionalidad permite a los usuarios docentes modificar los datos ingresados previamente en una clase virtual. Los datos a editar son los mismos citados en Agregar Clase.

6.3.1.2.4 Publicar Clase Virtual

Esta funcionalidad permite a los usuarios docentes hacer visibles las clases virtuales a los usuarios alumnos, los cuales podrán enterarse de su realización a través de los mensajes destacados de U-Cursos y podrán visitar su información.

6.3.1.2.5 Eliminar Clase Virtual

Esta funcionalidad permite a los usuarios docentes eliminar clases virtuales previamente agregadas. Cabe destacar que las clases eliminadas no podrán ser recuperadas con posterioridad.

6.3.2 Modelo de Datos

El modelo de datos del sistema muestra las tablas necesarias para almacenar la información como se muestra en la Figura 13.

Modelo de Datos Clases Virtuales

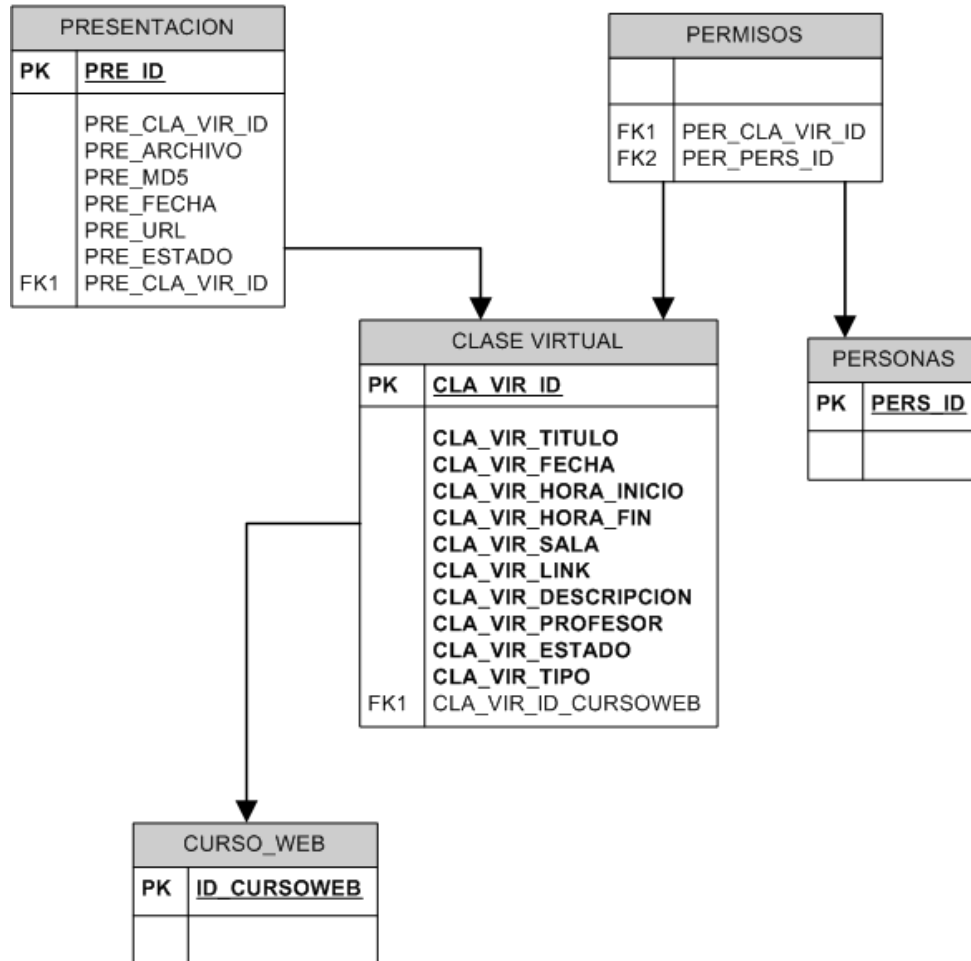


Figura 13 Modelo de Datos¹⁶

¹⁶ PK = Primay Key (Llave primaria), FK = Foreign Key (Llave foránea)

6.3.2.1 Especificación de Datos

El modelo de datos fue creado en base a los estándares de desarrollo de ADI, que cuenta con algunas particularidades:

- Nombres de atributos contienen de prefijo las 3 primeras letras de las palabras que conforman el nombre de la tabla, y como postfijo el nombre completo descriptivo del atributo.
- Las llaves foráneas contendrán como prefijo la regla anterior seguidas de los nombres de atributos de su tabla de origen.
- Fechas son almacenadas como tipo Timestamp.

6.3.2.1.1 Tabla CLASE VIRTUAL

Almacena los datos de las clases virtuales creadas en el modulo de U-Cursos del sistema. La descripción de sus datos puede verse en la Tabla 7.

Tabla 7 Detalle Tabla Clases Virtuales

Nombre Atributo	Descripción	Tipo/Tamaño	Null	Tipo de Atributo
CLA_VIR_ID	Identificador de la clase virtual	Integer(10)	No	Llave Primaria
CLA_VIR_TITULO	Título de la clase virtual	Varchar(255)	No	
CLA_VIR_FECHA	Fecha de Realización de la clase virtual	Integer(11)	No	
CLA_VIR_HORA_INICIO	Hora de inicio de la clase virtual ¹⁷	Time	No	
CLA_VIR_HORA_FIN	Hora de fin de la clase virtual	Time	No	
CLA_VIR_SALA	Nombre de la sala virtual ¹⁸	Varchar(255)	No	
CLA_VIR_LINK	Contiene el link de teletransportación a la sala virtual en Second Life	Varchar(255)	No	

¹⁷ Si bien el sistema maneja horarios predefinidos, el modelo de datos está diseñado para expandir estos a cualquier horario, con el reparo de resolver adecuadamente el tema de superposición de horarios agendados.

¹⁸ Por el momento existe solamente una sala disponible, pero el modelo de datos está diseñado pensando en el aumento de este tipo de salas.

CLA_VIR_DESCRIPCION	Contiene la descripción de la clase virtual	Varchar(255)	Si	
CLA_VIR_PROFESOR	Rut del docente que dicta la clase	Integer(11)	No	
CLA_VIR_ESTADO	Estado de la publicación de la clase, puede ser publicada/no publicada (con valores 1/0)	Integer(1)	No	
CLA_VIR_TIPO	Tipo de clase abierta o cerrada (con valores 1/0)	Integer(1)	No	
CLA_VIR_ID_CURSOWEB	Identificador global de curso en la plataforma U-Cursos.	Integer(11)	No	Llave Foránea

6.3.2.1.2 Tabla PERMISOS

Almacena la lista de usuarios permitidos para cada una de las clases privadas que se realicen en el sistema. La descripción de sus datos se puede ver en la Tabla 8.

Tabla 8 Detalle Tabla Permisos

Nombre Atributo	Descripción	Tipo/Tamaño	Null	Tipo de Atributo
PER_CLA_VIR_ID	Identificador de la clase virtual	Integer(11)	No	Llave Foránea
PER_PERS_ID	Identificador de una persona en la Plataforma U-Cursos (este valor corresponde al RUT de la persona)	Integer(11)	No	Llave Foránea

6.3.2.1.3 Tabla PRESENTACION

Almacena las imágenes de cada una de las slides que constituyen las presentaciones de las clases creadas. Estas imágenes son alojadas en el servidor de U-Cursos. La descripción de sus datos se puede ver en la Tabla 9.

Tabla 9 Detalle Tabla Presentación

Nombre Atributo	Descripción	Tipo/Tamaño	Null	Tipo de Atributo
PRE_ID	Identificador de la presentación	Integer(11)	No	Llave Primaria
PRE_CLA_VIR_ID	Identificador de la clase virtual a la que corresponde	Integer(11)	No	Llave Foránea
PRE_ARCHIVO	Nombre del archivo guardado	Varchar(255)	No	
PRE_MD5	Md5 del archivo guardado para comprobar consistencia	Varchar(32)	No	
PRE_FECHA	Fecha de subida al servidor de U-Cursos	Integer(11)	Si	
PRE_URL	Dirección URL del recurso	Varchar(255)	No	
PRE_ESTADO	Estado de la imagen, 1 es una presentación valida 0 es invalida	Integer(1)	No	

6.3.2.1.4 Tabla CURSO_WEB y PERSONAS

Estas tablas son parte de la plataforma U-Cursos, de los cuales se extrajeron los datos mencionados en las especificaciones de la Figura 13 con funciones disponibles en la API de U-Cursos. Los datos utilizados corresponden a las llaves foráneas explicadas en la Tabla 7 y Tabla 8.

6.3.3 Scripts de Procesos Batch U-Cursos

Estos procesos tienen como finalidad entregar la información requerida por los objetos del *Aula Virtual* dentro Second Life automáticamente y sin interacción de los usuarios del sistema.

Existen 2 scripts de procesos batch que de detallan a continuación.

6.3.3.1 Script alimentador de Slides

Este tiene como objetivo principal entregar la lista de URLs en orden, de cada una de las slides pertenecientes a la presentación del curso en que se dicta en el momento de la petición por parte del objeto *Presentador Web* que se encuentra en la sala virtual.

En el caso de que no existan presentaciones asociadas en el momento del requerimiento, este entrega por defecto la URL de U-Cursos para ser mostrada en el *Presentador Web*.

6.3.3.2 Script alimentador de Permisos

Este tiene 2 objetivos; el primero es informar al *Aula Virtual* dentro de Second Life si la clase actual es de carácter abierta o cerrada. En el caso de ser una clase abierta, se permite ingresar a todos los avatares que soliciten entrar, y en el caso de ser una clase cerrada, este script envía la lista de avatares que tienen permisos de ingresar a la clase, con lo cual el sistema de acceso podrá discriminar de forma adecuada.

6.3.4 Objetos Second Life

En Second Life todos los objetos físicos virtuales existentes son creados por los mismos residentes. Todos los objetos como autos, casas, joyas e inclusive pelo están compuestos de uno o más partes primitivas llamadas prims.

Un prim es un objeto geométrico básico tridimensional. Estos son como los ladrillos en Second Life y componen todos los objetos complejos. Los prims son completamente personalizables, y las herramientas de construcción permiten a los avatares definir la forma, color y textura de cada una de las caras del prim, como se muestra en la Figura 14.

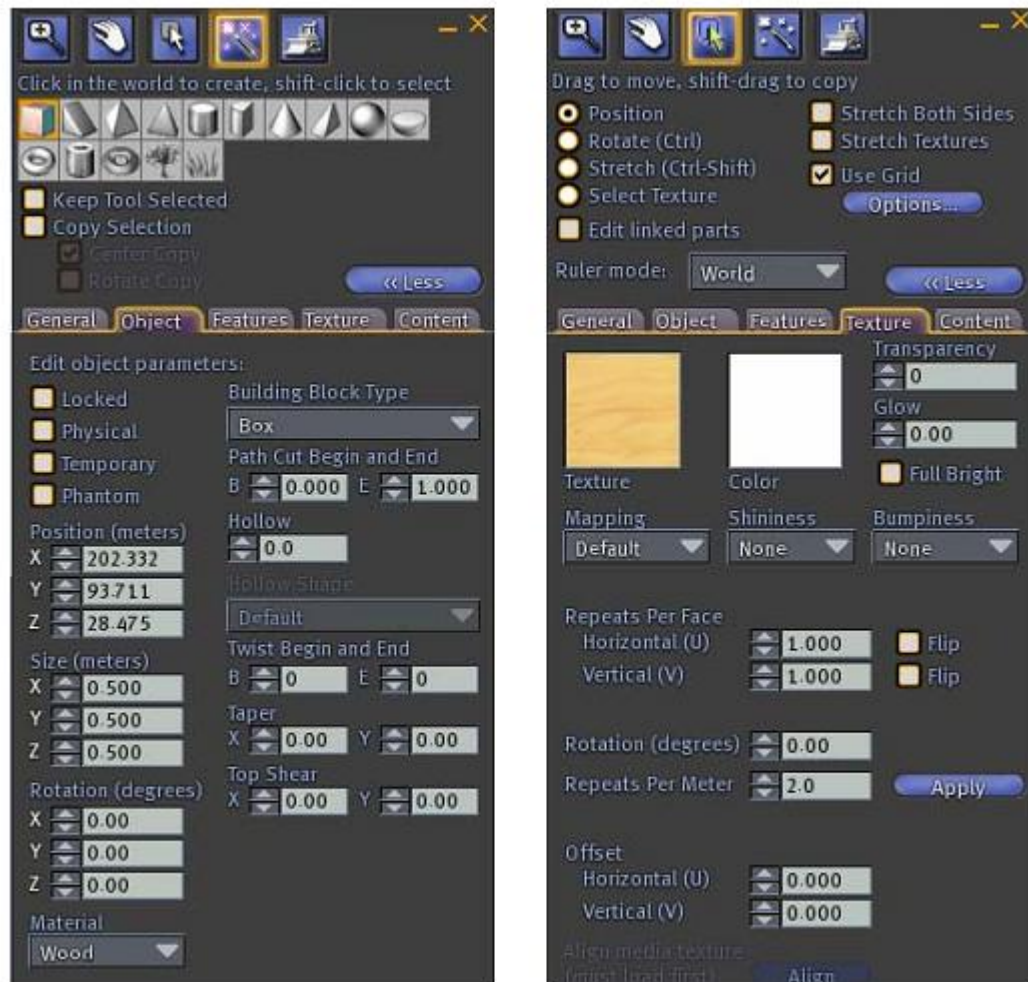


Figura 14 Herramientas de Construcción de Second Life

Dentro de un objeto compuesto por varios prims enlazados existe uno que es el principal llamado *root prim* o prim raíz y el resto son los prims hijos o *child prims*.

Los objetos también contienen comportamientos asociados, los cuales son programados a través del lenguaje de scripting inmerso en Second Life. Estos scripts deben ser adjuntados a los objetos, los cuales pasan a ser objetos interactivos. Para los objetos compuestos, estos scripts deben ser adjuntados al prim raíz.

Otra característica de los objetos en Second Life es que a las superficies de los prims se le pueden asociar texturas basadas en 2 tipos:

- Texturas de imágenes: Estas pueden ser subidas al inventario personal del avatar con un costo de 10L\$ cada una. Luego estas imágenes pueden ser asociadas tantas veces como se desee a los objetos creados, y se puede editar en tamaño, escala, color, etc.
- Texturas de contenido Web: Este tipo de textura esta solamente disponible para los avatares que poseen un terreno virtual en Second Life, pues junto con él se entrega la posibilidad de agregar contenido multimedia a los objetos que se encuentren en dicha región.

Además, existen algunas restricciones de construcción de objetos dentro de Second Life:

1. Los prims pueden ser de tamaño mayor o igual a 0.01 metro y menor o igual a 10 metros por cada uno de sus lados. Si se requieren prims de mayor tamaño deben utilizarse los megaprims.
2. Los objetos creados pueden ser compuestos por un número máximo de 256 prims.
3. Los prims, para poder enlazarse, deben estar separados por un máximo de 54 metros. Esta distancia depende de la posición y del tamaño de los lados del eje X/Y/Z de cada prim.

6.3.4.1 Objetos a construir

Se diseñaron un conjunto de diferentes objetos virtuales, los cuales están orientados para ser utilizados en las clases que se realicen en Second Life. Un resumen de estos es mostrado en la Tabla 10.

Tabla 10 Objetos Creados en Second Life

Objeto	Características
Sala virtual	Corresponde a una sala de clases virtual inspirado en el salón Gorbea de la FCFM, con las siguientes funcionalidades: <ul style="list-style-type: none">• Acceso diferenciado según tipo de clase• Interacción accesible y confortable para los auditores
Presentador de Diapositivas Web	Emulador de diapositivas (presentador de imágenes web) que consta con las siguientes opciones: <ul style="list-style-type: none">• Adelantar• Retroceder• Reiniciar
Paneles Informativos	Paneles informativos que entregan indicaciones, consejos y todo tipo de información a los avatares.

6.3.4.2 Sala Virtual

Esta corresponde a la principal construcción dentro de Second Life, tanto por la complejidad en su construcción, como por la importancia que cobra al alojar las clases que se realicen en su interior.

La creación de esta estructura fue inspirada en el Salón Gorbea de la FCFM Figura 15 y Figura 16, la cual cuenta con una tribuna para los asistentes y una tarima para el presentador. Un primer prototipo de esta fue realizado en “Google SketchUp” como se muestra en la Figura 17.



Figura 15 Auditorio Gorbea Tarima



Figura 16 Auditorio Gorbea Tribuna

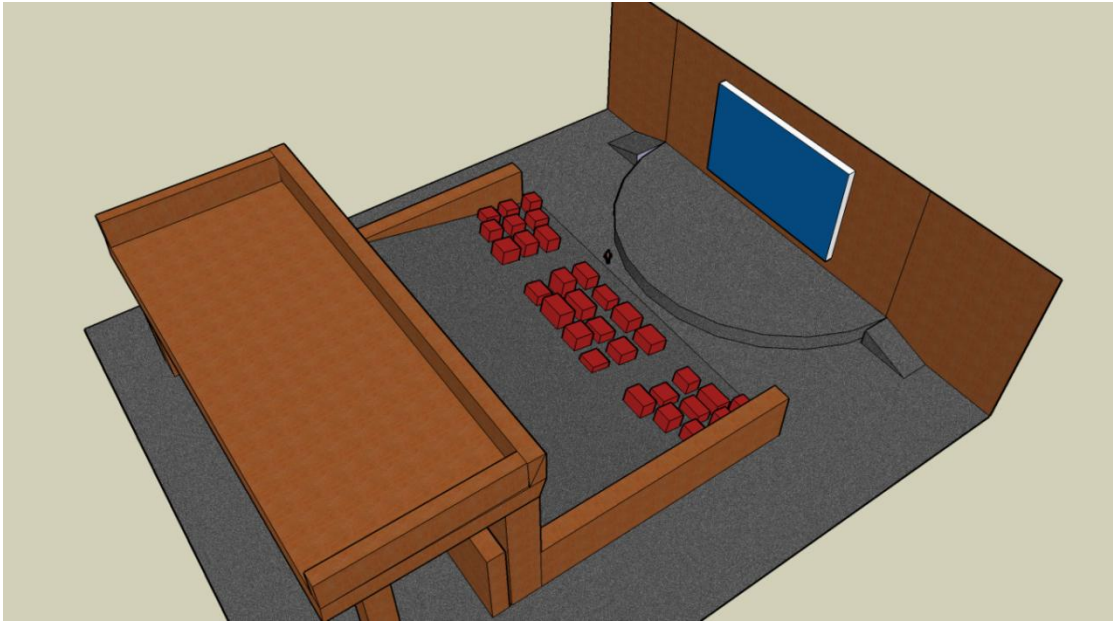


Figura 17 Prototipo Sala Virtual

6.3.4.3 Presentador de Diapositivas Web

Este objeto corresponde al segundo objeto más importante en el concepto de la clase virtual, pues permite a los profesores exponer los contenidos junto con sus diapositivas enviadas previamente mediante U-Cursos.

La creación de este objeto fue inspirada en la tecnología actual que existe en la escuela para el manejo de la presentación a través de un terminal y una pantalla gigante con la diapositiva. Un prototipo de éste puede verse en la Figura 18.

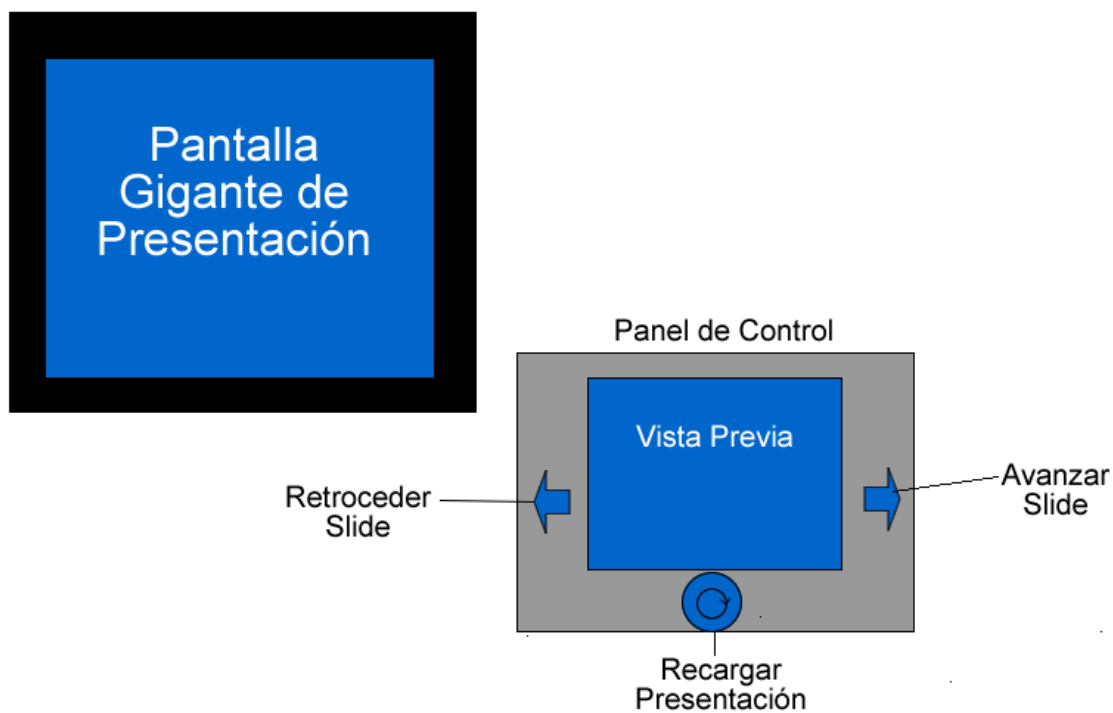


Figura 18 Prototipo Presentador Web

6.3.4.4 Paneles Informativos

Estos objetos permitirán entregar información a los avatares. La idea principal radica en dar mensajes o instrucciones útiles para la interacción con los objetos creados o para el desarrollo mismo de la clase.

Un símil en la realidad son los paneles electrónicos presentes en las estaciones de buses, aeropuertos, metro, o los paneles con mapas que se encuentran en la FCFM. Un prototipo de un panel informativo puede verse en la Figura 19.

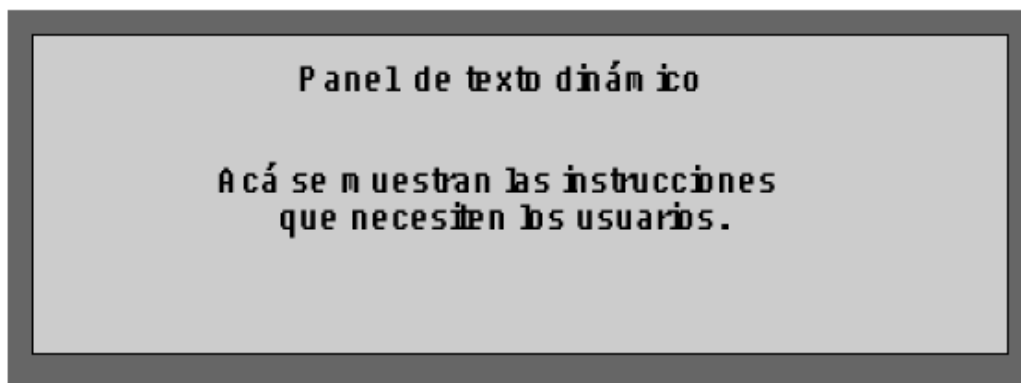


Figura 19 Prototipo panel informativo

7 Implementación

7.1 Aspectos Generales

7.1.1 Sistema parte U-Cursos

El sistema en la parte de U-Cursos fue desarrollado bajo los estándares de programación y seguridad impuestos por el ADI de la FCFM, basándose en el documento de “Buenas Costumbres” construido por dicha Área. En él destacan las siguientes normas:

- Todas las llamadas a la base de datos fueron encapsuladas en funciones utilizando la clase DB que provee PEAR.
- La separación entre la capa de control y de presentación o vista se llevó a cabo a través del uso de templates de la biblioteca PHPLib, adaptada a las necesidades de ADI.

El sistema fue construido en el servidor de desarrollo “adi2” con características similares a las del servidor de producción, el cual posee un sistema operativo basado en Linux con PHP versión 5.

7.1.2 Sistema parte Second Life

El sistema en la parte de Second Life fue desarrollado bajo las recomendaciones y consejos ilustrados por los tutoriales disponibles en la wiki oficial de Second Life¹⁹.

El cliente utilizado correspondió siempre a la última versión oficial disponible, la cual cambio 3 veces en el período de desarrollo sin afectar el transcurso del mismo.

El avatar personal utilizado para la creación de todos los objetos tiene como nombre *Nobu Xue*, el cual puede verse en la Figura 20.

¹⁹ <http://wiki.secondlife.com/>



Figura 20 Avatar Nobu Xue

7.2 Parte U-Cursos

7.2.1 MVC en U-Cursos

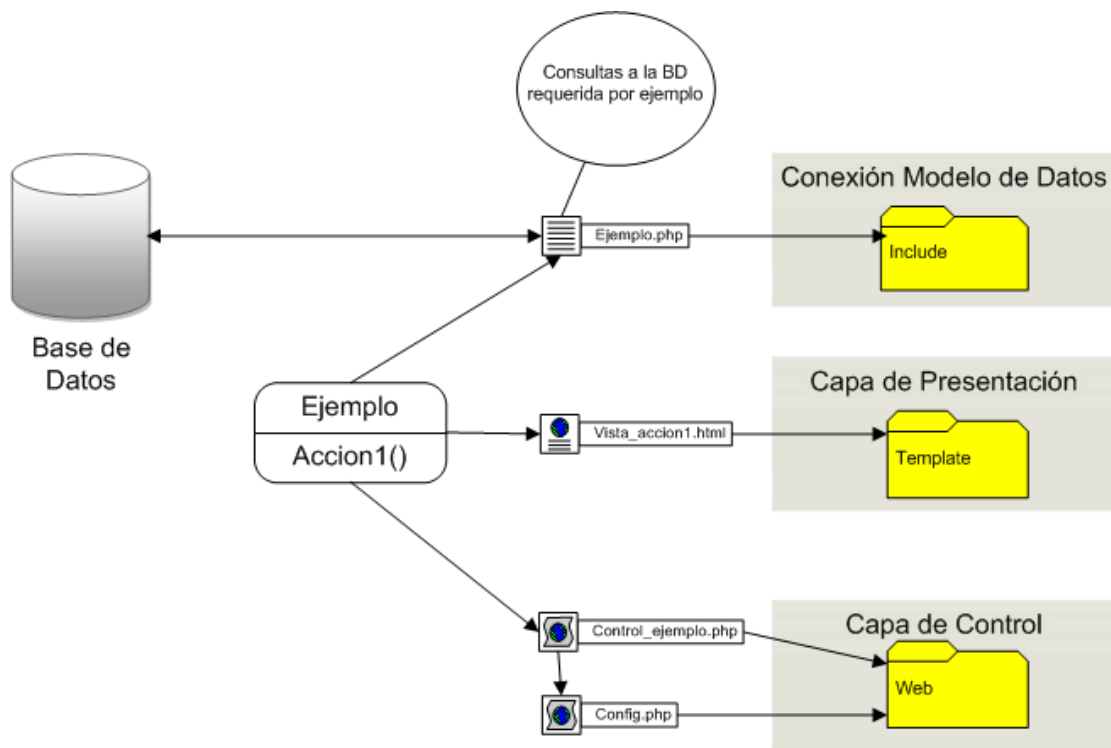


Figura 21 MVC en U-Cursos

El modelo vista controlador existente en U-Cursos se refleja en la estructura de carpetas requerido para la incorporación de nuevos módulos, como se muestra en la Figura 21, y que consta de las siguientes características:

- **Carpeta Include:** Esta carpeta contiene todos los scripts con las funciones propias del módulo. Aquí se realizan tanto las funciones de algoritmos necesarias, como las llamadas encapsuladas a la base de datos. Cada función realiza una tarea única y su nombre es representativo de la acción que realiza.
- **Carpeta Template:** Esta carpeta almacena todas las vistas del módulo, es decir los archivos HTML. Estos archivos tienen la característica de trazar sólo los elementos visuales recurrentes y comunes a todos los usuarios, pero además tienen la capacidad de permitir contenido dinámico que varía según el tipo de usuario o los permisos de éste, por esta razón son llamados plantillas o templates.
- **Carpeta Web:** Esta carpeta almacena los scripts que se acceden desde la red. Cada script representa una página Web del módulo, que controlan la petición de información realizada a la carpeta include, y la despliega según el trazado realizado en las plantillas de la carpeta template. Además en esta carpeta existe un archivo de configuración que debe llevar el nombre de *config.php*, el cual contiene los parámetros de configuración a la llave de InfoNúcleo además de los requerimientos a las librerías que se necesiten para el desarrollo.

7.2.2 Estructura del Módulo



Figura 22 Estructura de Carpetas

La estructura de carpetas del módulo corresponde a la mostrada en la Figura 22. La función de cada archivo es detallada en la Tabla 11.

Tabla 11 Estructura de Carpetas

Estructura de Carpetas	
Carpeta Include	
Nombre de Archivo	Características
Funciones.php	Contiene todas las consultas a la Base de Datos, de forma encapsulada. Además provee otras funciones que ayudan al procesamiento de información.
Carpeta Template	
Nombre de Archivo	Características
Agregar.html	Contiene la plantilla de visualización del formulario para agregar nuevas clases virtuales.
Clase.html	Contiene la plantilla de visualización del detalle de una clase virtual, con datos como los permisos de usuarios, slides de la presentación, sala virtual, etc.
Clases.html	Contiene la plantilla de visualización de la lista de todas las clases virtuales creadas en el módulo, y corresponde a la página principal del servicio.
Editar.html	Contiene la plantilla de visualización del formulario de edición de clases virtuales.
Carpeta Web	
Nombre de Archivo	Características
Agregar.php	Contiene la lógica del proceso de negocio para la creación de nuevas clases virtuales.
Clase.php	Contiene la lógica de visualización del detalle de las clases virtuales, para mostrar la información de forma adecuada.
Config.php	Contiene la configuración del módulo a la llave de InfoNúcleo del sistema y los requerimientos a todas las librerías que se necesitan.
Editar.php	Contiene la lógica del proceso de negocio para la edición de clases virtuales.
Eliminar.php	Contiene la lógica de negocios para el proceso de eliminación de clases virtuales.
Index.php	Contiene la lógica de negocios para visualizar la lista de clases virtuales creadas en el sistema, con las opciones adecuadas para los distintos usuarios.
Publicar.php	Contiene la lógica de negocios para publicar/despublicar las clases virtuales en el sistema.

7.2.3 Capa de Presentación

7.2.3.1 Navegación del Módulo

La navegación del módulo es muy similar en todos los servicios que se proveen en la plataforma U-Cursos. Esta es mostrada en la Figura 23 y corresponde a la navegación de un usuario docente. Para el caso de los usuarios alumnos, es la misma pero sólo pueden acceder a la función *Ver* al final del mapa.

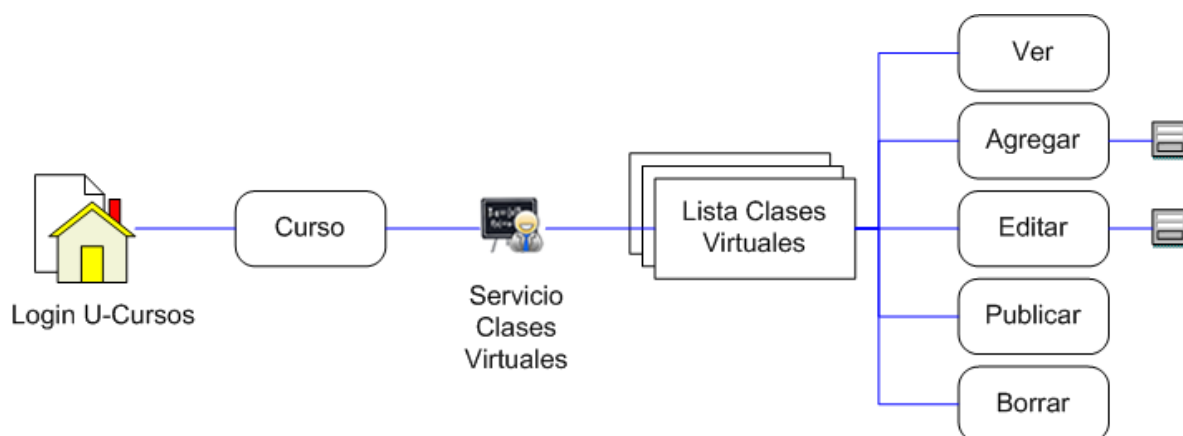


Figura 23 Mapa de Navegación Clases Virtuales

7.2.3.2 Interfaz de Usuario

La interfaz de usuario fue desarrollada bajo los patrones de diseño ya existentes en U-Cursos, los cuales vienen dados por las hojas de estilo de la plataforma.

El diseño de U-Cursos provee en todo momento un panel lateral izquierdo, que contiene datos del usuario, cursos actuales, comunidades e institución a las que pertenece. En la parte superior siempre encontramos las opciones y el buscador del sistema, y más abajo la lista de servicios contextuales a la sección que se visita, además de la ayuda del sistema. El área de trabajo de los servicios se encuentra en el centro de la pantalla, donde por lo general la información es desplegada en tablas y con las opciones de edición a la derecha, como se ve en la Figura 24.



Figura 24 Diagramación U-Cursos

Bajo este patrón se desarrollaron las diferentes pantallas de usuario del módulo clases virtuales.

7.2.3.2.1 Pantalla Principal Clases Virtuales

Es la primera pantalla a la que acceden indistintamente los permisos todos los usuarios del sistema, con la diferencia que los usuarios alumnos sólo pueden ver la lista de clases virtuales publicadas y tienen acceso a ver los detalles de éstas.

Por su parte, los usuarios docentes ven la lista de todas las clases creadas en el sistema, junto con las opciones de ver detalles, publicar/despublicar, agregar, editar y borrar clases virtuales. Una vista de esta pantalla es mostrada en la Figura 25.



adi2.ing.uchile.cl (157,7 kb.) » Inicio » Instituciones » Facultad de Cs. Físicas y Matemáticas » Cursos » El2001-23 Taller de Proyecto (30636) » [clases virtuales](#) » Clases Virtuales

clases virtuales *Beta!*

Título	Fecha	Horario	Estado	Publicada	Opciones
Proxima Clase	07/07/2009	10:15:00 - 11:45:00	En proceso	<input checked="" type="checkbox"/>	Editar Borrar
Clase test 2	23/06/2009	18:00:00 - 19:30:00	Realizada		
prueba de agregar clase	18/05/2009	16:00:00 - 18:00:00	Realizada		
Nueva clase	11/06/2009	18:00:00 - 19:30:00	Realizada		
prueba de subir clase	02/06/2009	08:30:00 - 10:00:00	Realizada		
Clase con avatar propio	28/05/2009	10:15:00 - 11:45:00	Realizada		
Prueba sin select sala	19/05/2009	18:00:00 - 19:30:00	Realizada		
Nueva clase	29/05/2009	14:30:00 - 16:00:00	Realizada		
Proxima Clase Privada	17/06/2009	14:00:00 - 16:00:00	Realizada		
Clase Prueba	15/06/2009	14:30:00 - 16:00:00	Realizada		
Clase TEst	27/06/2009	10:15:00 - 11:45:00	Realizada		
Nueva Clase	09/06/2009	18:00:00 - 19:30:00	Realizada		

Agregar >

Figura 25 Pantalla principal Clases Virtuales

7.2.3.2.2 Ver Detalles Clase Virtual

Esta pantalla permite visualizar la información de una clase virtual, y está disponible para todos los tipos de usuarios del sistema. Los datos de esta pantalla corresponden a datos estáticos, salvo el link de teletransporte a la sala virtual el cual abre la aplicación de Second Life y teletransporta al lugar indicado al Avatar. La presentación también queda disponible para ver cada una de las slides haciendo click sobre ellas. Una vista de esta pantalla se muestra en la Figura 26.



adi2.ing.uchile.cl (157,7 kb.) » Inicio » Instituciones » Facultad de Cs. Físicas y Matemáticas » Cursos » Et2001-23 Taller de Proyectos (30638) » clases virtuales » Detalles Clase Virtual

clases virtuales *Beta!*

Detalle Clase Virtual	
Título	Proxima Clase
Profesor	Javier R. Villanueva G.
Avatar	profesor1 Elan
Estado	En proceso
Fecha	07/07/2009
Horario	10:15:00 - 11:45:00
Sala Virtual	NeoGorbea Teletransportar
Descripción	clase numero 2
Tipo de Clase	Privada
Presentación	<ul style="list-style-type: none"> Slide 0

Asistentes Permitidos		
	Nombre	Avatar
Camila Aguirre C. switch		profesor1 Alex
Francisco Javier Barrientos Riveros switch		nobu Xue

Figura 26 Ver Detalles Clase Virtual

7.2.3.2.3 Agregar Clase Virtual

Esta pantalla corresponde al formulario de creación de clases virtuales. Sólo los usuarios docentes pueden acceder a esta función. Una vista de una clase abierta es mostrada en la Figura 27 y una clase privada en la Figura 28. Los detalles de cada campo pueden verse en la Tabla 12.

Tabla 12 Datos disponibles para agregar Clase Virtual

Dato	Observación
Título	Puede ser cualquier nombre con caracteres alfanuméricos con un límite de 255.
Profesor	El nombre del docente que crea y dicta la clase virtual.
Fecha	Provee un calendario para escoger la fecha en que se desea impartir la clase virtual.
Horario	Corresponde al horario del día en que se desea realizar la clase virtual, se encuentra en el formato de módulos horarios de la FCFM.
Presentación	Permita agregar slides de presentación para la clase. La restricción es que cada una de estas slides debe ser un archivo de imagen JPG o PNG.
Sala Virtual	El nombre de la sala disponible para realizar la clase virtual. Por el momento solo existe una.
Descripción	Permite insertar cualquier descripción basado en caracteres alfanuméricos con un largo máximo de 255.
Privacidad	Permite elegir uno de los 2 tipos de privacidad, en el caso de hacer una clase privada se despliega la lista de alumnos, los cuales pueden ser elegidos para su asistencia a la clase.

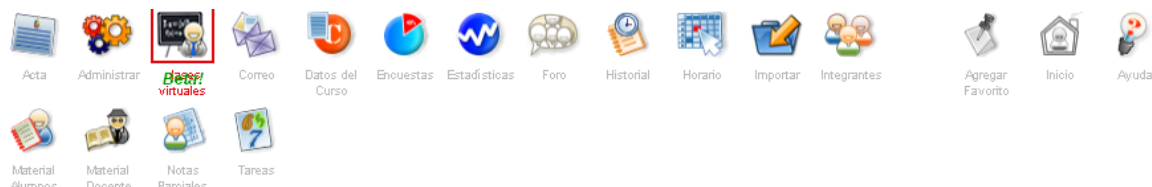


clases virtuales **Beta!**

Agregar Clase Virtual

Título	<input type="text"/>
Profesor	Javier R. Villanueva G.
Fecha	<input type="text" value="07/07/2009"/>
Horario	1 (08:30:00 - 10:00:00)
Presentación	<input type="text"/> <input type="button" value="Examinar..."/>
Sala Virtual	NeoGorbea
Descripción	<input type="text"/>
Privacidad Clase	<input checked="" type="radio"/> Abierta <input type="radio"/> Privada <i>Abierta: Cualquier usuario puede entrar a la clase.</i> <i>Privada: Sólo los usuarios seleccionados pueden entrar a la clase.</i>

Figura 27 Agregar Clase Virtual Abierta



adi2.lng.uchile.cl (157,6 kb.) » Inicio » Instituciones » Facultad de Cs. Físicas y Matemáticas » Cursos » E12001-23 Taller de Proyecto (30638) » clases virtuales » Nueva Clase Virtual

clases virtuales *Beta!*

Agregar Clase Virtual

Título	<input type="text"/>
Profesor	Javier R. Villanueva G.
Fecha	07/07/2009
Horario	1 (08:30:00 - 10:00:00)
Presentación	<input type="text"/> <input type="button" value="Examinar..."/>
Sala Virtual	NeoGorbea
Descripción	<div style="border: 1px solid gray; height: 100px;"></div>
Privacidad Clase	<input type="radio"/> Abierta <input checked="" type="radio"/> Privada <i>Abierta: Cualquier usuario puede entrar a la clase. Privada: Sólo los usuarios seleccionados pueden entrar a la clase.</i>
Asistentes	<ul style="list-style-type: none"> • <input type="checkbox"/> Seleccionar Todos • <input type="checkbox"/> Camila Aquirre C. switch • <input type="checkbox"/> Francisco Javier Barrientos Riveros switch • <input type="checkbox"/> Alfonso Javier Cornejo Castillo switch • <input type="checkbox"/> Thomas Dabovich K. switch • <input type="checkbox"/> Mauricio Daniel Dinamarca Caceres switch • <input type="checkbox"/> Sven Edvard Harfaagar Mandiola switch • <input type="checkbox"/> Eduardo Alejandro Jimenez Sanchez switch • <input type="checkbox"/> Jorge Osvaldo Martinez Villarroel switch • <input type="checkbox"/> Patricio Alfredo Perez Villanueva switch • <input type="checkbox"/> Javier Rozas V. switch • <input type="checkbox"/> Nicolas Ignacio Sanhueza Matamala switch • <input type="checkbox"/> Pablo Andres Tapia Caro switch • <input type="checkbox"/> Cristobal Andres Vera Toledo switch

Figura 28 Agregar Clase Virtual Privada

7.2.3.2.4 Editar Clase Virtual

Esta pantalla corresponde al formulario de edición de clases virtuales. La única diferencia con el formulario de agregar, es que este carga los datos previamente ingresados en el sistema en los campos correspondientes. Una vista de la pantalla de edición es mostrada en la Figura 29.

Material Alumnos, Material Docente, Notas Parciales, Tareas, Acta, Administrar, **clases virtuales**, Correo, Datos del Curso, Encuestas, Estadísticas, Foro, Historial, Horario, Importar, Integrantes, Agregar Favorito, Inicio, Ayuda

adi2.ing.uchile.cl (157,7 kb.) » Inicio » Instituciones » Facultad de Cs. Físicas y Matemáticas » Cursos » E12001-23 Taller de Proyectos (30638) » clases virtuales » Editar Clase Virtual

clases virtuales *Beta!*

Editar Clase Virtual

Título	Clase de muestra
Profesor	Javier R. Villanueva G.
Fecha	17/07/2009
Horario	5ª. (16:00:00 - 18:00:00)
Presentación	<input type="text"/> <input type="button" value="Examinar..."/>
Sala Virtual	NeoGorbea
Descripción	aquí van los detalles de la clase
Privacidad Clase	<input checked="" type="radio"/> Abierta <input type="radio"/> Privada <small>Abierta: Cualquier usuario puede entrar a la clase. Privada: Sólo los usuarios seleccionados pueden entrar a la clase.</small>

Figura 29 Editar Clase Virtual

7.3 Parte Second Life

La implementación de todos los objetos creados en Second Life fue realizado en 2 partes:

1. **Creación del objeto físico virtual:** Corresponde a la creación del objeto en base al enlazamiento de prims, elección de formas, texturas y colores.
2. **Creación de scripts de comportamiento:** Corresponde a la programación de los scripts que manejan el comportamiento de cada uno de los objetos, para la interacción con los avatares y procesos batch asociados a ellos.

7.3.1 Sala Virtual

Para la construcción de la sala se tomó en cuenta la restricción que se tiene de máximo de prims por isla²⁰, minimizando el uso de estos, simplificando los detalles de la sala y eliminando las estructuras innecesarias como la galería, y así evitando posibles problemas de lag en el servidor de Second Life. Para las estructuras más grandes se hizo uso de megaprims, obtenidos a través de otro avatar.

La estructura completa tiene un número de 291 prims, ocupando un volumen de 2000m³ (40 metros de largo, 50 metros de ancho y 10 metros de altura) y está compuesta por 6 unidades estructurales ensamblables, dada la restricción número 3 de construcción de objetos expuesta anteriormente. Las características de cada una de las partes se detalla a continuación:

1. **NeoGorbea Norte:** Contiene el muro norte, la tarima donde se posa el podio y el presentador de diapositivas. Consta de 16 prims.
2. **NeoGorbea Sur:** Contiene el muro Sur de la estructura, junto con la puerta de acceso y un teletransportador para ingresar a la sala.
La puerta de acceso a la sala contiene un script que responde a los avatares cuando presionan sobre ella. Este script²¹ realiza llamadas al servidor de U-Cursos y pregunta si la modalidad de la clase es abierta o privada, si la clase es abierta permite el acceso a todos los usuarios, sino esta sugiere a los usuarios a usar el teletransportador.

²⁰ Ver Anexo 10.2

²¹ Ver Anexo 10.3

El teletransportador es un objeto que pide la lista de avatares que pueden asistir a la clase, los cuales podrán ser teletransportados al interior de la sala. Si un usuario no tiene permisos de accesos, el teletransportador enviará un mensaje al avatar diciendo que no tiene los permisos suficientes.

La estructura completa consta de 14 prims.

3. **NeoGorbea Oeste:** Contiene el muro oeste del auditorium. Consta de 9 prims.
4. **NeoGorbea Este:** Contiene el muro este del auditorium. Consta de 10 prims.
5. **NeoGorbea Top:** Contiene el techo de la estructura. Consta de 8 prims.
6. **NeoGorbea Bottom:** Contiene el suelo de la estructura y la tribuna de asistentes con una capacidad máxima de 60 sillas. Cada una de las sillas consta de 4 prims (la relación óptima entre recursos y estética de un objeto), y de un script para que los asistentes puedan tomar asiento de forma fácil y correcta, es decir, al realizar click sobre la superficie de estas, permitiendo a los avatares tomar asiento sólo en la posadera de la silla. Esta estructura consta de 235 prims constituyendo la estructura más compleja de la sala, debido principalmente al número de sillas.

La Sala Virtual es el objeto en Second Life que tiene más elementos interactivos en su composición global, constando de los siguientes componentes:

1. **Sillas inteligentes:** Para la comodidad de los asistentes, y evitar el caos dentro del salón se creó un script asociados a cada una de las sillas de la tribuna, el cual permite a los avatares tomar asiento en las posaderas de estas, con vista hacia la tarima de presentación de la sala. Esta acción es desencadenada con un simple click en la silla en que se desea tomar asiento.
2. **Puertas deslizables:** Las puertas de la sala corresponde a una puerta deslizable de 2 hojas, las cuales al permanecer cerradas, se encuentran unidas y al abrirlas se desplazan ambas hojas con respecto al centro dando la sensación de apertura.
3. **Control de acceso:** Está compuesto por la puerta de la sala y el teletransportador que contienen los script de acceso que se conectan con el script *setpermisos.php* de U-Cursos, el cual entrega la información del tipo de clase actual y la lista de los usuarios permitidos. El objeto teletransportador, en el caso de una clase privada recibe la lista de avatares que pueden acceder a ella y los teletransporta al interior de la sala, si los avatares no están en la lista, este objeto informa sobre su restricción de permisos.

Algunas imágenes de la sala virtual pueden verse en la Figura 30, Figura 31, Figura 32, Figura 33 y Figura 34.

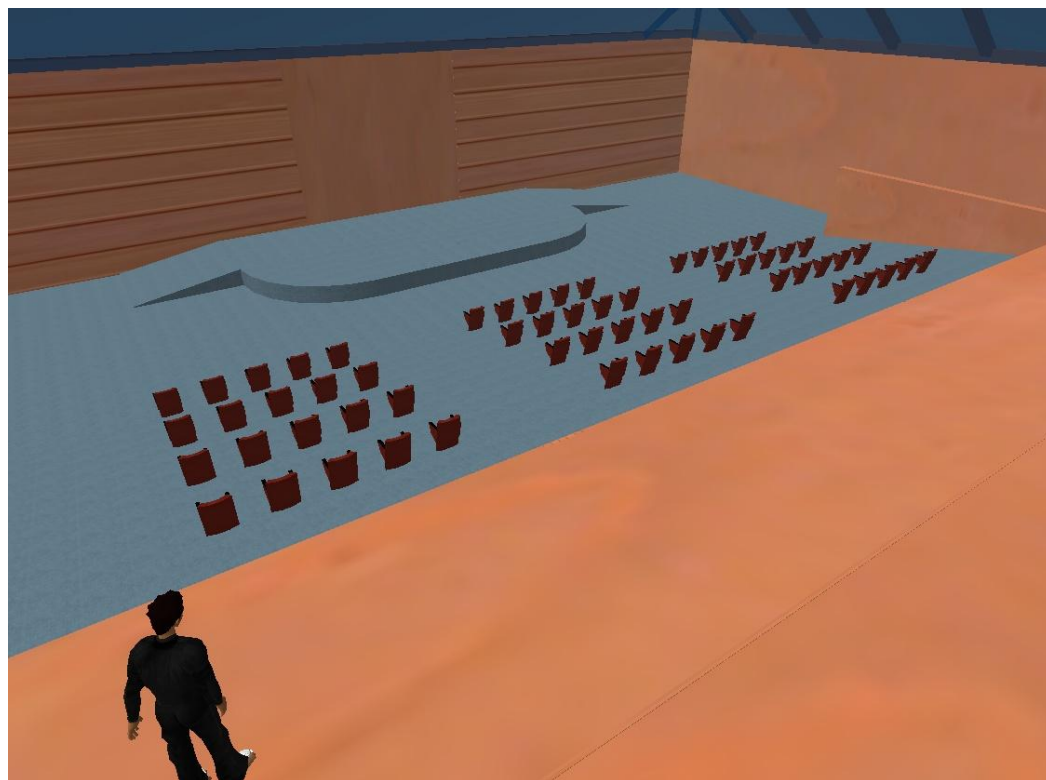


Figura 30 Sala Virtual vista de tribuna



Figura 31 Sala Virtual vista de sillas



Figura 32 Sala Virtual vista de tarima



Figura 33 Sala Virtual vista exterior



Figura 34 Sala Virtual Puerta y Teletransportador

7.3.2 Presentador de Diapositivas Web

El presentador de diapositivas web permite la visualización de diapositivas alojadas en el servidor de U-Cursos a través del manejo de listas de URLs que contienen cada una de las slides de la presentación.

El objeto presentador web consta de 3 componentes principales:

- **Panel de control:** Permite el manejo de la presentación por parte del docente, a través de los botones (avanzar, retroceder, reinicio) y muestra la imagen de la slide actual.
- **Podio de conferencias:** Posee el panel de control para el docente, además de un micrófono y el logo de la FCFM.
- **Pantalla gigante:** Realiza la exposición de la presentación a los asistentes a la clase (como los telones usados en las proyecciones mediante data shows).

La estructura final consta de 14 prims, y contiene un script²² que hace el manejo de la presentación a través del panel de control ubicado en el podio. Las funcionalidades que provee son las siguientes:

- **Reinicio de la presentación:** Al activar el botón de reinicio, el objeto realiza una petición al servidor de U-Cursos con la lista de URLs donde se encuentra cada slide de la presentación, para luego almacenarlas en una lista, y posiciona la pantalla del presentador con la primera slide de esta lista.
- **Avance de Slide:** Este botón avanza a la siguiente slide almacenada en la lista de la presentación. En el caso de encontrarse en la última slide, esta vuelve cíclicamente al inicio.
- **Retroceso de Slide:** Este botón retrocede a la slide anterior almacenada en la lista de la presentación. En el caso de encontrarse en la primera slide, esta muestra la última slide de la presentación.

²² Ver Anexo 10.4

Como se comentó anteriormente, al accionar el botón de *Reinicio* el objeto realiza una petición al script *setppt.php* el cual entrega la lista de URLs que contienen cada una de las slides de la presentación, las cuales son almacenadas en una lista. El accionar de los botones atrás y adelante del panel de control, lo que hacen es básicamente navegar esta lista de forma cíclica según las ordenes entregadas. El flujo de trabajo es el mostrado en la Figura 35.

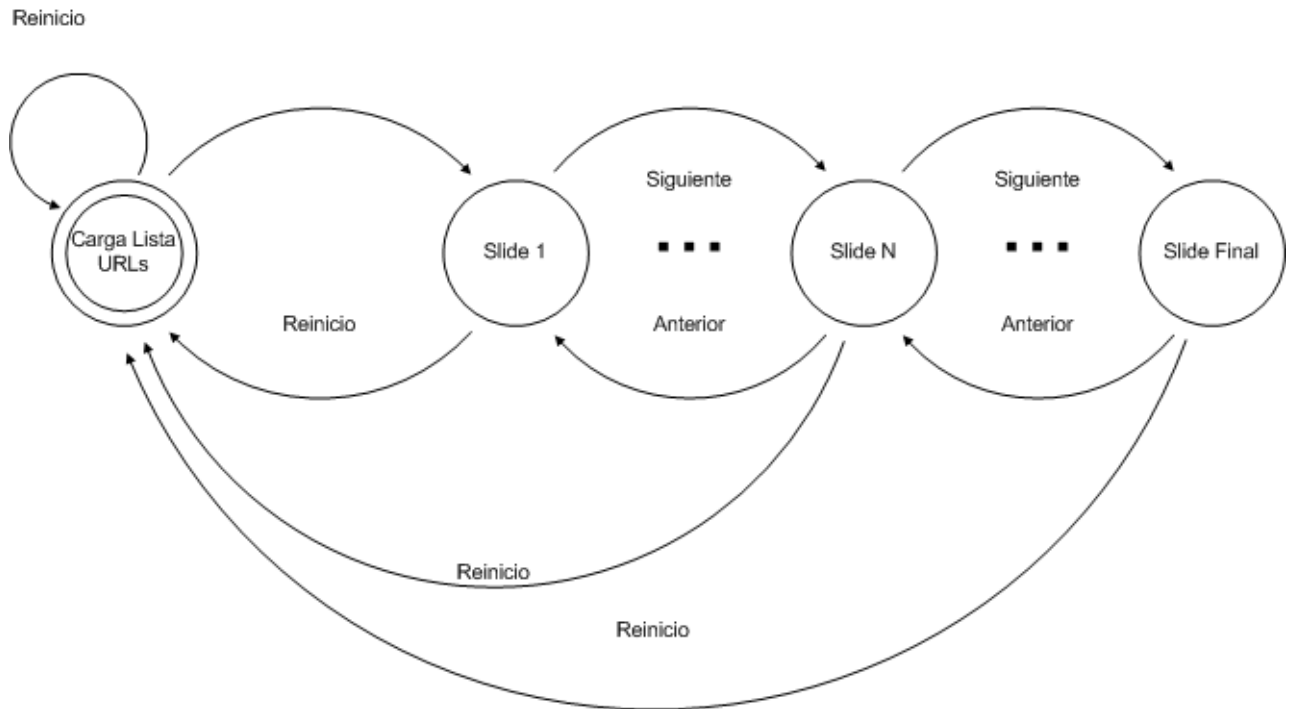


Figura 35 Flujo Presentador Web

El presentador Web creado y su panel de control respectivo pueden verse en la Figura 36 y la Figura 37 respectivamente.



Figura 36 Presentador Web

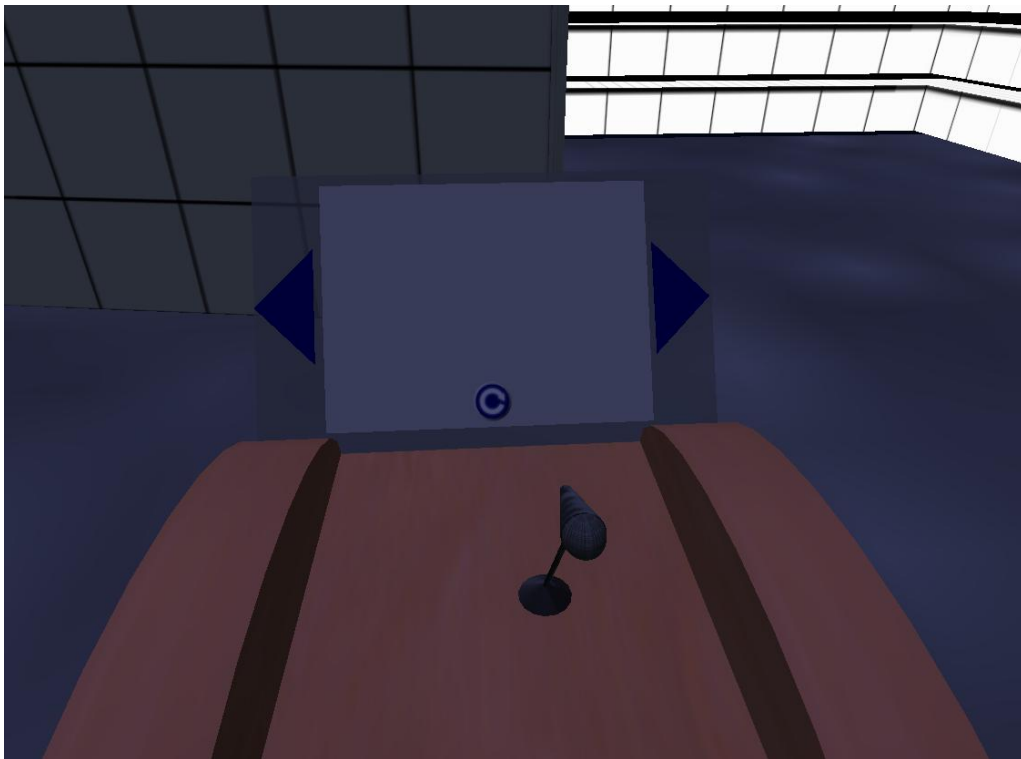


Figura 37 Panel de Control Presentador Web

7.3.3 Paneles Informativos

Se crearon una colección de paneles informativos que pueden ser dispuestos en diferentes partes de la sala virtual. Estos tienen como finalidad entregar instrucciones a los usuarios o cumplir un rol estético dentro del ambiente. Dependiendo de su naturaleza, los paneles creados se dividen en 2 tipos: paneles de texto dinámicos y paneles estáticos.

7.3.3.1 Paneles de texto dinámicos

Los paneles de texto dinámicos corresponden a objetos dentro de Second Life, los cuales tienen en alguna de sus superficies información escrita. Cabe destacar que el mostrar caracteres alfanuméricos escritos sobre la superficie de los objetos no es una cuestión trivial en Second Life, y para su realización se utilizó la librería “XyzzyText” disponible en la wiki²³ de Second Life [37]. Esta librería lo que hace básicamente es leer cada carácter de una textura, las cuales se despliegan en un prim especial logrando mostrar hasta 10 caracteres concatenados en una superficie dando la impresión de ser un panel electrónico. La concatenación de estos prims en un orden especial y cumpliendo un conjunto de reglas (un nombre especial para cada prim) permiten la creación de paneles con una capacidad variable de caracteres, y por lo tanto la creación de paneles informativos de largo variable. Para mayor detalle, ver el Anexo 10.5.

Para alimentar los paneles de texto se puede inyectar contenido de diferentes fuentes, las cuales por supuesto pueden cambiar en el tiempo modificando el texto de la superficie de los paneles, lo cual lo hace una característica muy interesante para entregar información a los usuarios.

²³ [http://wiki.secondlife.com/wiki/XyzzyText#XyzzyText - 10_Character](http://wiki.secondlife.com/wiki/XyzzyText#XyzzyText_-_10_Character)

Se diseñaron 2 diferentes versiones de estos paneles, dependiendo en la forma que reciben el texto para desplegar en su superficie:

- Paneles Informativos Vía NoteCard: El objeto obtiene el texto desde una NoteCard adjunta al objeto. Cuando el texto de esta NoteCard cambia, también lo hace el texto mostrado en la superficie.
El objeto está compuesto por un panel de 200 caracteres (4 columnas y 5 filas) y 3 scripts:
 - XyzyText10: script de mapeo de caracteres.
 - Slave: script de complemento para mayor performance.
 - Uinfo: script de manejo de texto y configuración de lectura desde NoteCard.
- Paneles Informativos Vía Web: El objeto obtiene el texto desde una dirección Web especificada en el script. El funcionamiento es muy parecido al objeto anterior, salvo el origen de su fuente.

Un ejemplo de este tipo de paneles puede verse en la Figura 38 y Figura 39.

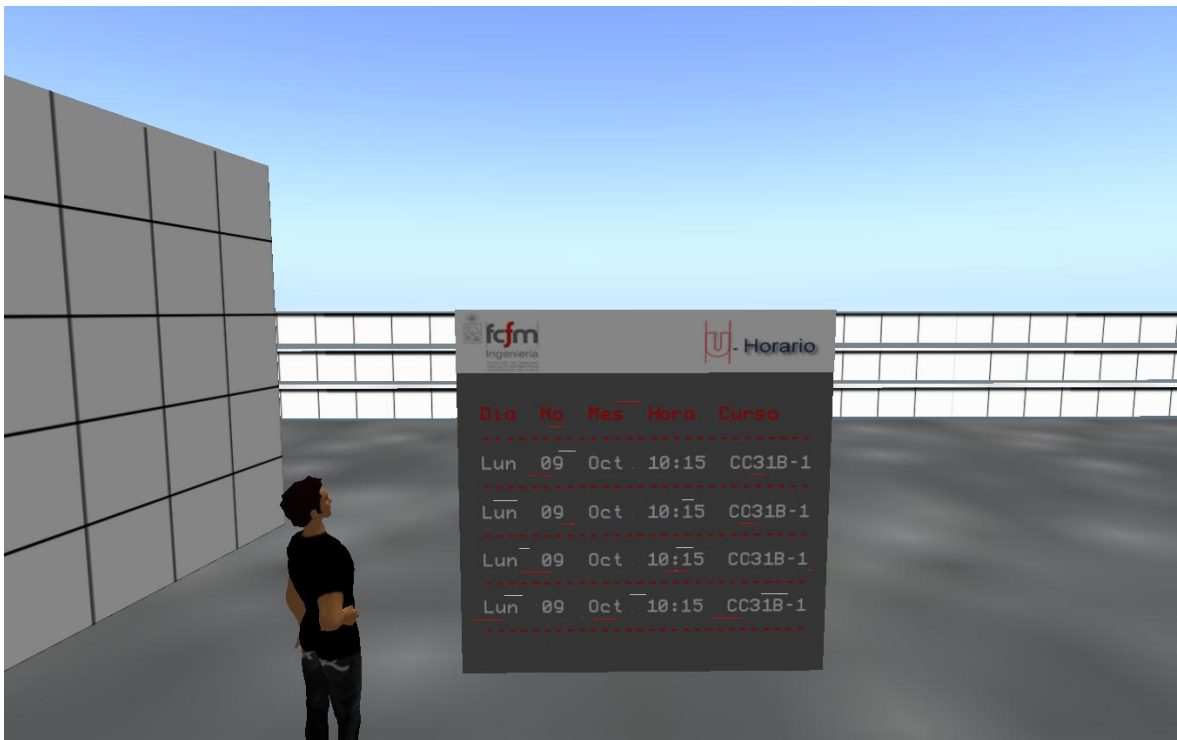


Figura 38 Panel de Texto Informativo 1



Figura 39 Panel de Texto Informativo 2

7.3.3.2 Paneles Estáticos

Los paneles estáticos son objetos inertes, es decir, sin scripts asociados, con la única característica que en una o más de sus caras presenta una imagen de textura de personalizada. Estas texturas fueron subidas al inventario y luego añadidas a los objetos mediante las herramientas de construcción que provee Second Life. Un ejemplo de estos paneles se muestra en la Figura 40.

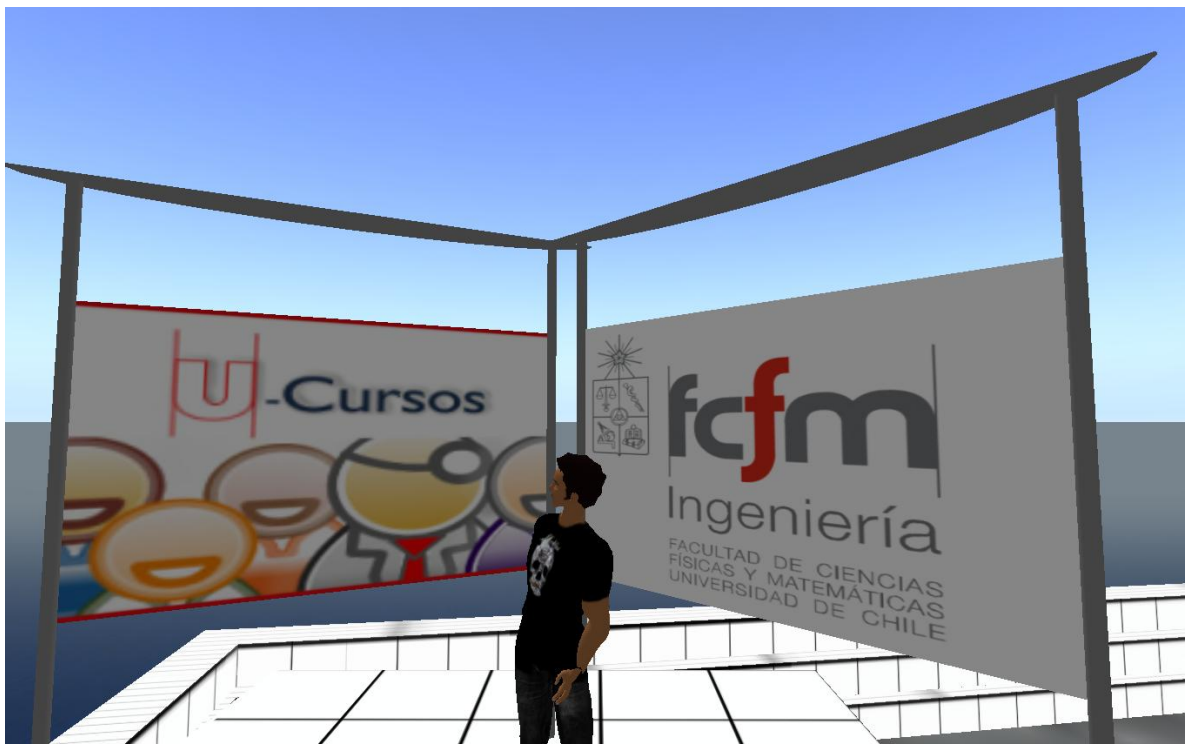


Figura 40 Paneles Estáticos Second Life

7.4 Parte Second Life U-Cursos

7.4.1 Procesos Batch U-Cursos

La plataforma U-Cursos provee un mecanismo de creación de procesos batch que pueden ser ejecutados de forma remota y externa a la plataforma conociendo los parámetros de ejecución correctos. Estos scripts deben ser almacenados en la carpeta *batch* del módulo en que se desea agregar.

Para la comunicación con Second Life se crearon 2 scripts batch que proveen diferentes características como se muestra en la Figura 41 y en la Tabla 13.

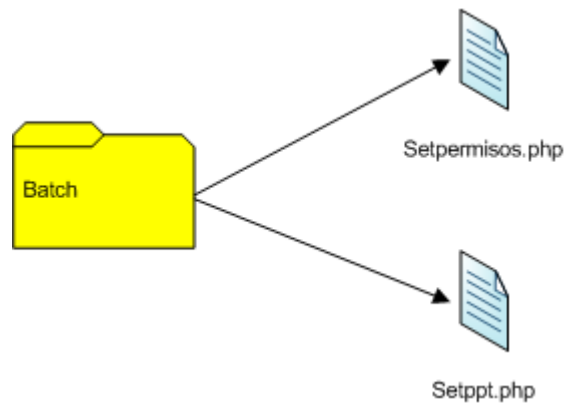


Figura 41 Scripts de procesamiento batch

Tabla 13 Detalle Procesos Batch

Script de Procesamiento Batch	Objeto en Second Life que se comunica	Descripción de la información transmitida	Acción en caso de no existir clase asignada
Setpermisos.php	Sala Virtual	El objeto Sala pregunta al script si la clase actual ²⁴ es abierta o privada, y en el caso de ser privada envía la lista de avatares que pueden ingresar a ella.	El objeto Sala permite el libre acceso a ella.
Setppt.php	Presentador de Diapositivas Web	El objeto Presentador pregunta al script cual es la clase actual y le pide la lista de URLs con las slides correspondiente a ella.	El Presentador muestra la página de inicio de U-Cursos

²⁴ Se entiende clase actual, como la que corresponde al momento de la pregunta por parte del objeto.

8 Conclusiones

Finalizadas las etapas de Diseño e Implementación del sistema, se dan por cumplidos los objetivos planteados en el presente trabajo.

Después de múltiples reuniones de control, revisión y pruebas de cada uno de los componentes, se logró la validación por parte del Área de Infotecnologías (ADI) del sistema en su totalidad. La integración del módulo de clases virtuales a la plataforma U-Cursos se hizo de forma exitosa y siguiendo todas las normas requeridas para tal efecto.

Las pruebas de asignación de clases virtuales, integración y comunicación entre Second Life y U-Cursos que se realizaron entregaron resultados satisfactorios. Si bien, no se alcanzó a la puesta en producción del sistema por razones de tiempo, se prevé que esta no debe tener problemas, salvo la redirección de las peticiones que se debe hacer por parte de los objetos en Second Life.

8.1 Contribuciones

La creación de este sistema permite a la FCFM expandir las fronteras de las aulas reales, para el proceso de enseñanza-aprendizaje. Los docentes y alumnos contarán con una herramienta opcional para concertar las clases diferente al contexto de una sala de clases física, ya sea por problemas de viajes de los docentes o por previo acuerdo de las partes involucradas.

La integración de las plataformas logradas abre una ventana al campo de la educación virtual, permitiendo generar instancias de trabajo e investigación de nuevos medios para el proceso de aprendizaje y enseñanza.

En términos de innovación, consolida el uso de la plataforma U-Cursos en su rol de apoyo y solución a problemas contingentes en temas de docencia.

8.2 Limitaciones

Una de las limitaciones actuales del sistema es que sólo provee 1 sala virtual para toda la FCFM, lo cual podría generar problemas de asignación de ésta para diferentes cursos en un mismo horario.

La sala virtual está pensada para realizar clases con el paradigma de presentador y auditores, lo cual en el mundo virtual de Second Life es algo restrictiva, pues uno de los motores de los mundos virtuales en el ámbito de la enseñanza es potenciar el aprendizaje construyendo, y participando activamente de este proceso. Sin embargo a medida que se necesite se pueden ir cambiando y adaptando los componentes creados para hacer las clases de forma más libre.

Las presentaciones que permite el sistema sólo son vía imágenes lo cual no es lo más cómodo para los docentes, pues requiere un pequeño trabajo de exportar sus versiones originales de diapositivas a imágenes y luego subirlas a U-Cursos, esto viene dado la imposibilidad técnica de trabajar con los formatos originales de diapositivas por parte de Second Life.

La sala provee de un máximo de 60 asistentes, los cuales según Second Life es el número máximo aceptable por región para no experimentar problemas de Lag. Por esta razón es deseable que la sala virtual nunca llegue a su límite natural y en el caso de requerir una clase para una gran cantidad de asistentes, se recomienda dividir a la audiencia en 2 o más grupos y realizar el mismo número de clases para los grupos creados.

Second Life, por todas las capacidades que presenta, no es una interfaz de usuario adecuada para la exposición de todo tipo de contenidos académicos, por lo cual esta solución no promete ser la solución para la educación a distancia, sino que representa una de las alternativas que estarán disponibles para los usuarios de la FCFM a la cual podrán optar. Para reforzar esta idea, en la actualidad en la FCFM existen cursos como el de Taller de Proyecto [38] que utiliza esta plataforma para su realización, lo que constituye un buen ejemplo de la utilidad de este sistema.

8.3 Trabajo Futuro

El modelo actual de clases en Second Life considera la utilización solamente de una sala, sin embargo el sistema fue pensado en la escalabilidad de este, y es posible ampliar el número de salas virtuales disponibles, si el uso de este recurso es demandado con mayor frecuencia.

Las asignaciones de horarios de petición de la sala virtual, para evitar el problema de choque entre horas, fue resuelto incipientemente utilizando el modelo de módulos horarios utilizados en la FCFM. Queda como trabajo futuro realizar una expansión de estos módulos a horarios de clases variables con la solución pertinente al problema de choques de horario.

La sala virtual se comporta como una sala real en términos de asignación de clases, por lo cual es posible estandarizar su asignación en el modulo de administración de salas y horarios de la plataforma U-Cursos.

Queda también como trabajo futuro la creación de diferentes objetos interactivos a disposición de la comunidad universitaria, para la realización de simulaciones de situaciones reales y creación de contenido educativo complementario a los ya existentes para aumentar las capacidades y posibilidades de enseñanza en el aula virtual.

También es posible la expansión de este sistema a las diferentes instituciones que utilizan U-Cursos, para lo cual en caso que estas lo requieran se podrá habilitar su uso.

9 Bibliografía

- [1] U-Cursos <https://www.u-cursos.cl/>
- [2] Cristián Céspedes, Julio Salas, “U-Cursos: Una plataforma de apoyo a la docencia presencial”, SOCHEDI 2006.
- [3] Cristián Céspedes, Manuel Ortega, Julio Salas, Javier Villanueva, “U-Cursos 3.0: Un enfoque hacia los asuntos comunitarios y las facilidades de uso”, SOCHEDI 2007.
- [4] Manuel Ortega, Julio Salas, Pedro Vargas, Javier Villanueva, “U-Grupos: Una herramienta de trabajo integrada a U-Cursos”, SOCHEDI 2008.
- [5] Second Life <http://secondlife.com/>
- [6] The Consultant-E online training and development consultancy - Education <http://www.theconsultants-e.com/>
- [7] Virtual Worlds – Best Practices in Education <http://www.vwbpe.org/>
- [8] Ana Eva Iribas Rudin, “Enseñanza virtual en Second Life: Una opción on-line animada para las universidades y las artes”, IV Jornada Campus Virtual UCM, pág. 125-142, Junio 2007.
- [9] Megan S. Coklin, “101 Uses for Second Life in the College Classroom”, Elon University Department of Computing Sciences, Febrero 2007.
- [10] Andrea de Lucia, Rita Francese, Ignazio Passero, Genoveffa Tortora, “Development and Evaluation of a virtual campus on Second Life: The case of Second DMI”, Dipartimento di Matematica e Informatica, Universidad of Salerno, Italia, Agosto 2008.
- [11] Club Penguin <http://www.clubpenguin.com/>
- [12] Habbo Hotel <http://www.habbo.es/>
- [13] World of Warcraft <http://www.worldofwarcraft.com/>
- [14] Nictropolis <http://www.nick.com/nicktropolis/game/>
- [15] Meez <http://www.meez.com/>
- [16] Zwinky <http://www.zwinky.com/>
- [17] Roma Victor <http://www.roma-victor.com/>
- [18] Alice <http://www.alice.org/>
- [19] Twinity <http://www.twinity.com/>
- [20] Project Wonderland <https://lg3d-wonderland.dev.java.net/>
- [21] Google Lively <http://www.lively.com/>
- [22] Active Worlds <http://www.activeworlds.com/edu/>
- [23] Open Simulator <http://opensimulator.org/>
- [24] Croquet <http://www.opencroquet.org/>
- [25] Qwaq <http://www.qwaq.com/>
- [26] There <http://www.there.com/>

- [27] Mariona Grané i Oró, Miguel Ángel Muras López, “Second Life, entorno virtual, aprendizaje real”, III Congreso Online – Observatorio para la CiberSociedad, 2006. <http://www.cibersociedad.net/congres2006/gts/comunicacio.php?id=851>
- [28] LSL Wiki <http://lslwiki.net/>
- [29] Heaton Research Online Books and Tutorials for Second Life
<http://www.heatonresearch.com/taxonomy/term/7>
- [30] Kline Labs. Tutorials <http://www.klinelabs.com/>
- [31] Online Free Script Library <http://www.freescrptlibrarysl.co.uk/>
- [32] A view from another life <http://secondview.wordpress.com/>
- [33] Second Life Wiki <http://wiki.secondlife.com/>
- [34] Second Life Support Center <https://support.secondlife.com/>
- [35] Sim Teach: Information and Community for Educators using Multi-User Virtual Environments <http://www.simteach.com/>
- [36] Área de Infotecnologías, “Documentación Plataforma U-Cursos”, 2005.
- [37] Biloshitskaya, Uliana, “Second Life Heads up display for University of Portsmouth”, Bsc thesis, University of Portsmouth UK, 2008.
- [38] Patricio Poblete, Ximena Vargas, Sergio Celis, Rocio Duque, “Concebir, diseñar, implementar y operar proyectos de Ingeniería en la realidad virtual: Aplicaciones en Second Life”, Universidad de Chile, Julio 2008.

10 Anexos

10.1 Formato de Horarios de la FCFM

Tabla 14 Formato de Horarios de la FCFM

Nombre	Hora de Inicio	Hora de Fin
1	08:30	10:00
1*	08:00	10:00
2	10:15	11:45
2*	10:00	12:00
3	12:00	13:30
3*	12:00	14:00
4	14:30	16:00
4*	14:00	16:00
5	16:15	17:45
5*	16:00	18:00
6	18:00	19:30
6*	18:00	20:00
7	19:30	21:00
7*	20:00	22:00

10.2 Límites de Prims por Región

Tabla 15 Límites de prims por región

Fracción de la Isla	Tamaño de parcela en m ²	Prims permitidos por parcela
1/128	512	117
1/64	1024	234
1/32	2048	468
1/16	4096	937
1/8	8192	1875
¼	16384	3750
½	32768	7500
Completa	65536	15000

10.3 Script de Puerta de Acceso

```
0001 string URL = "http://adi.ing.uchile.cl/~jbauza/permisos.php"; //URL ejemplo de script con permisos
0002 key http;
0003 list users;
0004 integer hasMoved = FALSE;
0005 float delay = 9.0;
0006 float moveBy = 3;
0007 integer CMD_MOVE = 1;
0008 integer CMD_RESTORE = 2;
0009 integer CLASS_TYPE = 1;
0010 integer leftDoor = 3;
0011 integer rightDoor = 2;
0012 string openSound = "cb340647-9680-dd5e-49c0-86edfa01b3ac";
0013 string closeSound = "e7ff1054-003d-d134-66be-207573f2b535";
0014
0015
0016 getAuth()
0017 {
0018     http = llHTTPRequest(URL, [HTTP_METHOD, "GET"], "");
0019 }
0020 }
0021 integer checkAuth(string agent)
0022 {
0023     integer index = llListFindList(users, [agent]);
0024     if(index!=-1)
0025     {
0026         return TRUE;
0027     }
0028     else
0029     {
0030         return FALSE;
0031     }
0032 }
0033 }
0034
0035 closeDoor()
0036 {
0037     llSetTimerEvent(0.0);
0038     llTriggerSound(closeSound, 1.0);
0039     llMessageLinked(LINK_ALL_OTHERS, CMD_RESTORE, "", NULL_KEY);
0040     hasMoved = FALSE;
0041 }
0042
```

```

0043 openDoor ()
0044 {
0045     llTriggerSound(openSound, 1.0);
0046     llMessageLinked(LINK_ALL_OTHERS, CMD_MOVE, (string)moveBy, NULL_KEY);
0047     hasMoved = TRUE;
0048 }
0049
0050 default
0051 {
0052     touch_start(integer count)
0053     {
0054         getAuth();
0055         string avatar = llDetectedName(0);
0056         if (CLASS_TYPE)
0057         {
0058             if (hasMoved)
0059             {
0060                 closeDoor();
0061             }
0062             else
0063             {
0064                 openDoor();
0065                 llSetTimerEvent(delay);
0066             }
0067         }
0068         else if(checkAuth(avatar))
0069         {
0070             llSay(0, "Clase Cerrada, intente ingresar con el Teletransportador");
0071         }
0072         else
0073         {
0074             llInstantMessage(llDetectedKey(0), "No tiene permisos para ingresar a la sala");
0075         }
0076     }
0077
0078     timer()
0079     {
0080         closeDoor();
0081     }
0082     http_response(key request_id, integer status, list metadata, string body)
0083     {
0084         if( request_id == http )
0085         {
0086             if( body=="1" ){
0087                 users = [];
0088                 CLASS_TYPE = 1;
0089             } else {
0090                 users = llParseString2List(body, [";"], []);
0091                 CLASS_TYPE = 0;
0092             }
0093         } else {
0094             llSay(0, "Intentenlo nuevamente");
0095         }
0096     }
0097
0098 }

```

10.4 Script de Presentador Web

```
0001 string URL = "http://adi.ing.uchile.cl/~jbauza/urlppt.php"; //url de ejemplo para obtener la lista de slides
0002 key http;
0003 integer total = 0;
0004 integer current = 0;
0005 list urls;
0006
0007 loadppt(){
0008     llParcelMediaCommandList([PARCEL_MEDIA_COMMAND_STOP]);
0009     http = llHTTPRequest(URL, [HTTP_METHOD, "GET"], "");
0010 }
0011 default
0012 {
0013     state_entry()
0014     {
0015         loadppt();
0016         current = 0;
0017     }
0018 }
0019
0020 http_response(key request_id, integer status, list metadata, string body)
0021 {
0022     if(request_id == http )
0023         urls= llParseString2List(body,[";"],[]);
0024     else
0025         llWhisper(0,"Fallo en la conexion al traer las ppt");
0026 }
0027
0028 on_rez(integer start_param)
0029 {
0030     llResetScript();
0031 }
0032
0033 touch_start(integer total_number)
0034 {
0035     total = llGetListLength(urls) - 1;
0036     string button = llGetLinkName(llDetectedLinkNumber(0));
0037     list buttonlist = ["reload", "back", "next"];
0038     list test = [(button)];
0039     integer index = llListFindList(buttonlist, test);
0040     if(index == -1)
0041     {
0042     }
0043     if(button == "back")
0044     {
0045         if(current > 0)
0046             current = current - 1;
0047         else
0048             current = total;
0049     }
0050     if(button == "reload")
0051     {
0052         loadppt();
0053         current = 0;
0054     }
0055     if(button == "next")
0056     {
0057         if(current < total)
0058             current += 1;
0059         else
0060             current = 0;
0061     }
0062 }
0063
0064 string webpage = llList2String(urls, current);
0065 llParcelMediaCommandList([PARCEL_MEDIA_COMMAND_URL,webpage]);
0066 llParcelMediaCommandList([PARCEL_MEDIA_COMMAND_PLAY]);
0067 }
0068 }
```


10.5 Uso librería XyzzyText

La librería permite la utilización la lista de caracteres mostrada en la Figura 42.

```
! " # $ % & ' ( )
* + , - . / 0 1 2 3
4 5 6 7 8 9 : ; < =
> ? @ A B C D E F G
H I J K L M N O P Q
R S T U V W X Y Z [
\ ] ^ _ ` a b c d e
f g h i j k l m n o
p q r s t u v w x y
z { | 0 ~ π θ x ° √
```

Figura 42 Caracteres XyzzyText

La librería está compuesta de 3 scripts principales:

1. **Script de configuración de prim:** Este se debe poner en un prim para transformarlo de tal forma que permita la visualización de hasta 10 caracteres en una de sus caras. Luego de esto el script se borra automáticamente del objeto.
2. **Script principal:** Este debe ser adjuntado al prim raíz del objeto. Este es responsable de configurar las texturas de caracteres que se muestran en la superficie del objeto. Otra característica de este es que puede manejar un conjunto de prims de texto enlazados, permitiendo la concatenación de texto con un largo superior a 10 caracteres, para lo cual el script conoce el orden de cada uno de los prims con texto. Estos prims deben llevar el nombre de “xyzzytext-numerodelinea-numerodecolumna”, por ejemplo para un panel de 60 caracteres de 2 filas, los nombres de los 6 prims de texto deben ser los mostrados en la Figura 43.

```
xyzzytext-0-0  xyzzytext-0-1  xyzzytext-0-2
xyzzytext-1-0  xyzzytext-1-1  xyzzytext-1-2
```

Figura 43 Orden de nombres para un panel de 60 caracteres

3. **Script esclavo:** Este permite dividirse la carga de trabajo con el script principal de mapeo de caracteres en el panel completo, asegurando una mayor rapidez a la solución.

Los pasos a seguir para la construcción de un panel con texto es el siguiente:

1. Crear un prim
2. Adjuntar el script de configuración al objeto creado para transformarlo
3. Hacer cuantas filas y columnas de los prims de texto creado con los nombres correspondientes.
4. Crear un panel que sirva de fondo al panel de texto y enlazar los prims, asegurándose que el prim de fondo sea el prim raíz.
5. Adjuntar al prim raíz el script principal y el esclavo.
6. Crear un archivo de control de texto para establecer el mensaje a mostrar. Un ejemplo de cómo manejar el control de texto es mostrado en la Figura 44.

```
integer DISPLAY_STRING      = 204000;
integer DISPLAY_EXTENDED  = 204001;
integer REMAP_INDICES     = 204002;
integer RESET_INDICES    = 204003;
integer SET_FADE_OPTIONS  = 204004;
integer SET_FONT_TEXTURE  = 204005;
integer SET_LINE_COLOR    = 204006;
integer SET_COLOR         = 204007;
integer RESCAN_LINKSET   = 204008;

integer gToggle;

default
{
    state entry()
    {
        lllisten(0, "", NULL_KEY, "");
    }

    listen(integer channel, string name, key id, string message)
    {
        if (gToggle)
            llMessageLinked(LINK_THIS, DISPLAY_STRING, name + ":" + message, "0");
        else
            llMessageLinked(LINK_THIS, DISPLAY_STRING, name + ":" + message, "1");

        gToggle=!gToggle;
    }
}
```

Figura 44 Ejemplo Control de Texto con librería XyzzyText