



UNIVERSIDAD SIMÓN BOLÍVAR
DECANATO DE ESTUDIOS PROFESIONALES
COORDINACIÓN DE INGENIERÍA DE LA COMPUTACIÓN

EXTENSIÓN DE UN SISTEMA DE GESTIÓN DE APRENDIZAJE CON UN MÓDULO DE ENCUENTROS PRESENCIALES

Por:

José Ricardo Pascarella Quijada

INFORME DE PASANTÍA

Presentado ante la Ilustre Universidad Simón Bolívar
como requisito parcial para optar al título de
Ingeniero en Computación

Sartenejas, Septiembre de 2017



UNIVERSIDAD SIMÓN BOLÍVAR
DECANATO DE ESTUDIOS PROFESIONALES
COORDINACIÓN DE INGENIERÍA DE LA COMPUTACIÓN

EXTENSIÓN DE UN SISTEMA DE GESTIÓN DE APRENDIZAJE CON UN MÓDULO DE ENCUENTROS PRESENCIALES

Por:

José Ricardo Pascarella Quijada

Realizado con la asesoría de:

Prof. Federico Flaviani

Ing. Christian Ament

INFORME DE PASANTÍA

Presentado ante la Ilustre Universidad Simón Bolívar
como requisito parcial para optar al título de
Ingeniero en Computación

Sartenejas, Septiembre de 2017

RESUMEN

Este documento presenta detalladamente el proceso de desarrollo de una extensión para un Sistema de Gestión de Aprendizaje (SGA) que permite el soporte de encuentros presenciales o seminarios entre estudiantes e instructores.

El módulo apoya a tres tipos de usuarios en la gestión de los encuentros. Los administradores, que crean los cursos y asignan los instructores. Los instructores que pueden administrar la asistencia y calificación; por último, los estudiantes que seleccionan cursos a los que asistir.

Para la realización de este módulo se usó la metodología de desarrollo de *software* ágil e iterativo por fases *scrum*. Para el desarrollo se utilizó el conjunto de soluciones informáticas Windows, Apache, MS SQL Server y PHP (WAMP), que consiste en Windows como plataforma de sistema operativo, Apache para el servicio web, *Structured Query Language*. Lenguaje de consulta estructurado (SQL) Server con gestor de bases de datos y el lenguaje multipropósito PHP.

Se logró mediante este proyecto que el SGA base de la compañía contenga la nueva funcionalidad que permite ser extendida y personalizada dependiendo de las necesidades de distintos clientes. Además, el módulo se implantó con éxito en la arquitectura de un cliente que previamente poseía el servicio de SGA.

Palabras clave: SGA, PHP: *Hypertext Preprocessor* (PHP), desarrollo, seminario.

ÍNDICE GENERAL

Introducción	1
Antecedentes	1
Planteamiento del problema	1
Justificación e importancia	2
Objetivos	2
Objetivo general	2
Objetivos específicos	3
1. Entorno Empresarial	4
1.1. Fischer, Knoblauch & Co.	4
1.2. Estructura organizacional	5
1.3. Cargo ocupado por el pasante	6
2. Marco Teórico	7
2.1. Conceptos básicos sobre el área de trabajo	7
2.1.1. <i>E-learning</i> o aprendizaje electrónico	7
2.1.2. <i>B-learning</i> o aprendizaje híbrido	8
2.1.3. Sistema de Gestión de Aprendizaje	8
2.1.4. Modelo Vista Controlador	9
2.1.5. Arquitectura cliente-servidor	10
2.1.6. Framework o entorno de trabajo	10
2.1.7. Manejador de bases de datos	10

3. Marco Tecnológico	11
3.1. Cliente	11
3.1.1. HTML	11
3.1.2. CSS	11
3.1.3. JavaScript	11
3.1.4. AJAX	12
3.1.5. JQuery	12
3.1.6. Bootstrap	12
3.2. Servidor	12
3.2.1. PHP	12
3.2.2. Smarty	13
3.2.3. Microsoft SQL Server	13
3.2.4. Servidor HTTP Apache	13
3.2.5. Swift mailer	14
3.3. Pruebas	14
3.3.1. PHP Unit	14
4. Marco Metodológico	15
4.1. Naturaleza del proyecto	15
4.2. Metodología ágil	15
4.3. <i>Scrum</i>	16
4.3.1. Dueño del Producto o <i>Product owner</i>	16
4.3.2. Equipo	16
4.3.3. Facilitador o Scrum master	17

4.3.4. Stakeholders o Partes interesadas	17
4.3.5. Eventos	17
4.3.6. Sprint	17
4.3.7. Sprint <i>Planning</i>	18
4.3.8. <i>Daily Scrum</i>	18
4.3.9. Sprint <i>Review</i>	18
4.3.10. Sprint <i>Retrospective</i>	18
4.3.11. Artefactos	19
4.3.12. <i>Product Backlog</i>	19
4.3.13. Sprint <i>Backlog</i>	19
4.4. Pruebas de <i>software</i>	19
5. Funcionalidades desarrolladas del sistema	20
5.1. Actores	21
5.1.1. Administrador	21
5.1.1.1. Gestionar seminarios	21
5.1.1.2. Gestionar ubicaciones	21
5.1.1.3. Gestionar sesiones	21
5.1.1.4. Notificar usuarios	22
5.1.2. Instructor	22
5.1.2.1. Listar sesiones	22
5.1.2.2. Imprimir lista de estudiantes	22
5.1.2.3. Evaluar estudiantes	22
5.1.3. Aprendiz	22

5.1.3.1. Gestionar reservas	23
6. Solución del problema	25
6.1. Módulo ubicaciones	25
6.2. Módulo Cursos del tipo Seminario	26
6.3. Usuario instructor	27
6.4. Sesiones de un seminario	28
6.5. Usuario aprendiz	30
6.6. Calificación de las sesiones	31
6.7. Sesiones de tipo <i>online</i> para los seminarios	32
6.8. Notificaciones	33
6.9. Estadísticas	33
6.10. Adaptación de los módulos al SGA de Bibliomed	34
7. Desarrollo de las funcionalidades	36
7.1. Primer Sprint	36
7.1.1. Objetivos	36
7.1.2. Actividades	36
7.2. Segundo sprint	38
7.2.1. Objetivos	38
7.2.2. Actividades	38
7.3. Tercer sprint	39
7.3.1. Objetivos	39
7.3.2. Actividades	39
7.4. Cuarto sprint	39

7.4.1. Objetivos	39
7.4.2. Actividades	40
7.5. Quinto sprint	40
7.5.1. Objetivos	40
7.5.2. Actividades	41
7.6. Sexto sprint	41
7.6.1. Objetivos	41
7.6.2. Actividades	42
7.7. Séptimo sprint	43
7.7.1. Objetivos	43
7.7.2. Actividades	43
7.8. Octavo sprint	44
7.8.1. Objetivos	44
7.8.2. Actividades	44
7.9. Noveno sprint	45
7.9.1. Objetivos	45
7.9.2. Actividades	45
7.10. Décimo sprint	46
7.10.1. Objetivos	46
7.10.2. Actividades	47
7.11. Actividades extra	47
7.11.1. Exportar preguntas de un SGA	47
7.11.2. Migración a UTF-8	48
7.11.3. Funcionalidad en modulo <i>responsive</i>	48

7.11.4. Imágenes de las vistas de login	48
7.12. Dificultades encontradas	49
Conclusiones y Recomendaciones	50
A. Diagramas	56
B. Screenshots de los sistemas	60

ÍNDICE DE FIGURAS

1.1. Estructura organizacional de Fischer, Knoblauch & CO (FKC).	5
5.1. Diagrama de casos de uso.	24
A.1. Diagrama que demuestra los distintos estados en los que se puede encontrar una sesión presencial.	56
A.2. Diagrama UML parcial de la base de datos del SGA de FKC previa al proyecto.	57
A.3. Diagrama UML parcial de la base de datos final del SGA de FKC.	58
A.4. Diagrama UML parcial de la base de datos final del SGA de Bibliomed. . .	59
B.1. Vista del listado de Ubicaciones.	60
B.2. Vista de la edición de una ubicación.	61
B.3. Vista de la creación de un seminario.	62
B.4. Vista del listado de cursos del sistema para el administrador.	63
B.5. Vista de la asignación de un seminario a un grupo.	63
B.6. Vista de la creación de una sesión de un seminario.	64
B.7. Vista del listado de las sesiones de un seminario.	65
B.8. Vista de los cursos de un instructor.	65
B.9. Archivo PDF generado con la lista de los estudiantes de un curso.	66
B.10. Vista de los cursos disponibles para el usuario aprendiz.	67
B.11. Vista de los cursos disponibles para el usuario aprendiz antes de la extensión realizada en la pasantía.	67
B.12. Vista del modal mostrado con los datos de la ubicación de la sesión correspondiente.	68

B.13.Vista de una sesión confirmada por el usuario.	68
B.14.Formato de los correos enviados por el sistema.	69
B.15.Vista de las estadísticas mostradas al administrador.	69
B.16.Vista de la adaptación hecha para la creación de seminarios en el SGA de Bibliomed.	70
B.17.Vista de cursos disponibles para el aprendiz antes de la integración de el módulo de seminarios en el SGA de Bibliomed.	71
B.18.Vista de una sesión reservada en el SGA de Bibliomed.	72

SIGLAS

- AJAX** *Asynchronous JavaScript And XML*. JavaScript y XML asíncronos. 11, 28, 52
- API** *Application Programming Interface*. Interfaz de programación de la aplicación. 9, 37, 38, 52
- ASP** *Active Server Pages*. 46
- CRUD** *Create, Read, Update, Delete*. Creación, Lectura, Actualización. 25, 27, 37, 39
- CSS** *Cascading Style Sheets*. Plantillas de estilos en cascada. 10, 11, 31, 36
- FKC** Fischer, Knoblauch & CO. 1–4, 15, 24, 35, 46, 47, 49, 50
- FTP** *File Transfer Protocol*. Protocolo de transferencia de archivos. 45, 50
- HTML** *Hypertext Markup Language*. Lenguaje de marcado de hipertexto. 10–12, 25, 45, 52
- HTTP** *Hypertext Transfer Protocol*. Protocolo de transferencia de Hipertexto. 12, 26, 30, 39, 48
- JSON** *JavaScript Object Notation*. Notación de Objetos de JavaScript. 28
- LMS** *Learning Management System*. Sistema de Gestión de Aprendizaje. 6, 7
- MVC** Modelo Vista Controlador. 7, 49, 51
- NPM** *NodeJS Package Manager*. Manejador de paquetes de NodeJS. 50
- PDF** *Portable Document Format*. Formato de documento portable. 31, 40
- PHP** *PHP: Hypertext Preprocessor*. 11–13, 24, 49
- SGA** Sistema de Gestión de Aprendizaje. 1, 2, 14, 19, 24, 35, 47, 49, 52
- SQL** *Structured Query Language*. Lenguaje de consulta estructurado. 9, 12, 36, 46, 49
- URL** *Unified Resource Locator*. Identificador de recursos unificado. 25, 38, 42
- UTF-8** *Unicode Transformation Format 8-Bit*. Formato de transformación de 8 bits. 47
- XML** *Extensible Markup Language*. Lenguaje de marcado extensible. 11

INTRODUCCIÓN

El presente documento describe el proceso de desarrollo del módulo de encuentros presenciales para un Sistema de Gestión de Aprendizaje (SGA) para Fischer, Knoblauch & CO (FKC) como proyecto de pasantías realizado por su autor. La compañía posee un sistema base que se personaliza y despliega según las necesidades de sus clientes.

El módulo permite a los administradores del sistema la creación de seminarios a los que le son asignados instructores y locaciones que definen el encuentro presencial. La gestión de las locaciones también debió ser desarrollado por el pasante.

Una vez definido un encuentro presencial, los usuarios que participan como aprendices en el sistema pueden inscribirse en estos eventos para luego ser evaluados en el dominio del material.

El pasante debió además integrar este desarrollo en el sistema de gestión de aprendizaje de uno de los clientes que previamente eran poseedores del producto de FKC, Bibliomed.

Antecedentes

El SGA básico de FKC es un producto que soporta las funciones básicas de cualquier sistema de gestión aprendizaje simple. Como lo son: manejo de usuarios y contenidos; seguimiento del proceso de aprendizaje, evaluaciones y herramientas de comunicación como foros y mensajes privados. Este sistema es personalizado e instalado generalmente en la infraestructura del cliente según sus necesidades.

Planteamiento del problema

Un sistema de gestión de aprendizaje se mantiene en constante actualización. A medida que las técnicas de enseñanza cambian, el sistema debe modificarse para dar el soporte necesario a las nuevas tendencias.

Al pasante entonces se le asignó la realización de un módulo que permita la gestión de seminarios presenciales en el sistema base que ofrece FKC.

El SGA de FKC no posee soporte para la organización de eventos presenciales en los

que participen alumnos e instructores. Esta forma de enseñanza es útil para la transmisión de conocimientos que no son fáciles de explicar a través de material multimedia.

Esta es una funcionalidad altamente requerida por los clientes de la compañía que permite al producto de FKC ser mas completo sobre todo al ser comparado con los sistemas de la competencia.

De lo expuesto anteriormente se identificó la necesidad de extender el sistema de FKC con un módulo que le permita a sus clientes gestionar encuentros presenciales entre instructores y estudiantes. Al mismo tiempo este módulo debía adaptarse a la estructura de cursos y grupos previamente existente en el sistema, tratando de incluir la menor complejidad posible para su futura extensión y personalización.

Para lograr esto, es necesario un profundo entendimiento del producto, con el fin de desarrollar un módulo que mantenga el mismo estilo tanto en la programación, como en el funcionamiento y la interfaz.

Justificación e importancia

Entre las funciones convencionales de los SGA actuales se encuentra el manejo de encuentros presenciales, funcionalidad considerada como necesaria por actuales y potenciales clientes. Siendo estos encuentros de importancia crítica para el flujo del conocimiento de mediana a alta complejidad que no puede ser expresado fácilmente por medio de cursos en línea. Por lo tanto, es de interés para la compañía poseer esta funcionalidad en su sistema.

Objetivos

A continuación, se exponen los objetivos generales y específicos que se buscan alcanzar en este desarrollo con la finalidad de contextualizar al lector respecto al informe de este proyecto de pasantía.

Objetivo general

El objetivo general de este proyecto es desarrollar un módulo que permita extender el sistema de gestión de aprendizaje de FKC y lo acerque a contener las funciones convencionales de los sistemas en la actualidad y que al mismo tiempo posea la flexibilidad de

ser personalizado para las distintas necesidades de los clientes. Que se ofrezca también la opción de configurar un SGA sin incluir el módulo en cuestión.

Objetivos específicos

- Desarrollar un módulo que permita el manejo de las locaciones donde se imparten los seminarios, manejado por el administrador.
- Extender la base de datos para el soporte del módulo.
- Integrar el nuevo tipo de curso a las funcionalidades previas de administrador sobre otros tipos de cursos, como estadísticas y asignación a grupos.
- Crear un nuevo tipo de usuario, instructor, que tenga potestad sobre los seminarios.
- Diseñar y desarrollar las interfaces necesarias para gestionar el nuevo módulo.
- Integrar el módulo tanto en el sistema base de FKC como en el sistema vendido previamente a Bibliomed, uno de los clientes que posee un SGA.

CAPÍTULO 1

ENTORNO EMPRESARIAL

En este capítulo se describe el entorno empresarial en el cual tuvo lugar el desarrollo del proyecto de pasantía, la empresa Fischer, Knoblauch & CO (FKC) filial de Frankfurt.

1.1. Fischer, Knoblauch & Co.

Es un proveedor de servicios multimedia especializado en el área de aprendizaje electrónico. Está presente en Frankfurt y Munich en Alemania así como en Basel, Suiza. Fundada en 1996 por Guy Fischer y Thomas Knoblauch.

Proveen consultoría en la integración y ampliación del aprendizaje electrónico a compañías de diversos sectores en Alemania y Bélgica. Se encargan de sugerir la elección de tecnologías, concepción del plan de aprendizaje, didácticas y metodología de la enseñanza, producción del contenido audiovisual, hasta la integración de la solución en el ambiente del cliente.

Si una compañía requiere enseñar una cierta habilidad a sus empleados contacta a un proveedor de servicios de aprendizaje electrónico, FKC los ayuda a integrar un plan aprendizaje a su empresa, que se ven materializados en entrenamientos basados en la web.

FKC también posee un Sistema de Gestión de Aprendizaje, la pieza de *software* en la que el pasante trabajó, que es personalizable y permite la organización de estos entrenamientos creados por la empresa.

Además, FKC haciendo uso de su departamento gráfico y programadores, también provee servicios de posicionamiento empresarial en la web, mediante la creación de páginas, logos y demás contenido multimedia que la compañía requiera.

Sus programadores día a día se enfrentan con diversos retos informáticos en distintos lenguajes de programación. Estos pueden ser: migraciones de sistemas de bases de datos, internacionalización de sus aplicaciones que llegan a estar hasta en diez lenguajes distintos, diseño de soluciones multiplataforma y el manejo e instalación de *frameworks* que faciliten la construcción de soluciones multimedia.

1.2. Estructura organizacional

En la figura 1.1 se muestra la estructura organizacional de FKC Frankfurt:

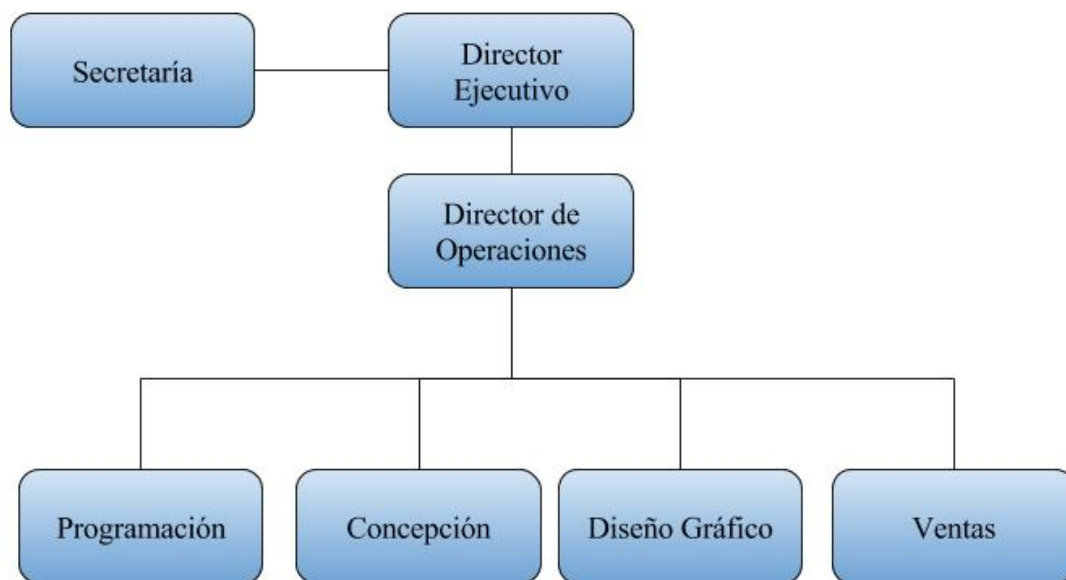


Figura 1.1: Estructura organizacional de FKC.

FKC Frankfurt es un equipo multidisciplinario donde es importante la comunicación entre los distintos *stakeholders*. Los creadores de concepto se unen a los diseñadores y los programadores para plasmar fielmente los requerimientos del cliente y obtener como resultado una solución hecha a la medida.

A continuación una descripción breve de los cargos en el organigrama.

Director Ejecutivo es la persona de máxima autoridad, encargada de la gestión y dirección administrativa en la organización.

Director de Operaciones es el responsable del control de las actividades diarias de la corporación y de manejo de las operaciones. Reporta directamente al director ejecutivo.

Secretaría encargada de dar apoyo a los empleados de la empresa en cuanto a la gestión de papeleo y la comunicación de las actividades.

La cuadrilla de programación se encarga de materializar las peticiones que llegan de los demás departamentos mediante soluciones informáticas.

El grupo de diseño gráfico se encarga de generar el material multimedia junto con los integrantes del grupo de concepción y los clientes.

El departamento de ventas se encarga del *marketing* de los productos ofrecidos por FKC.

1.3. Cargo ocupado por el pasante

El pasante perteneció al grupo de programación que se muestra en la figura 1.1 donde formó parte de un equipo de 6 programadores.

CAPÍTULO 2

MARCO TEÓRICO

En el presente capítulo es presentado y descrito el conjunto de conceptos, términos y pilares teóricos relevantes sobre los cuales se basó este proyecto.

2.1. Conceptos básicos sobre el área de trabajo

2.1.1. *E-learning* o aprendizaje electrónico

Se denomina aprendizaje electrónico (conocido también por el anglicismo *e-learning*) a la educación a distancia completamente virtualizada a través de los nuevos canales electrónicos (las nuevas redes de comunicación, en especial Internet), utilizando para ello herramientas o aplicaciones de hipertexto (correo electrónico, páginas web, foros de discusión, mensajería instantánea, plataformas de formación, etc.) como soporte de los procesos de enseñanza-aprendizaje[19].

Gracias a las nuevas tecnologías de la información y la comunicación (TIC), los estudiantes “en línea” pueden comunicarse y colaborar con sus compañeros “de clase” e instructores (profesores, tutores, mentores, etc.), de forma síncrona o asíncrona, sin limitaciones espacio-temporales. Es decir, se puede entender como una modalidad de aprendizaje dentro de la educación a distancia en la que se utilizan las redes de datos como medios (Internet, intranets, etc.), las herramientas o aplicaciones hipertextuales como soporte (por ejemplo, correo electrónico, web, chat, etc.) y los contenidos y/o unidades de aprendizaje en línea como materiales formativos. Como ejemplo, simples imágenes, audio, video, documentos, llegando hasta complejas producciones multimedia.

Las ventajas que ofrece la formación en línea son las siguientes:

Eliminación de barreras espaciales y temporales (desde su propia casa, en el trabajo, en un viaje a través de dispositivos móviles, etc.). Supone una gran ventaja para actores geográficamente dispersos o alejados.

Prácticas en entornos de simulación virtual, difíciles de conseguir en formación presencial, sin una gran inversión.

Enriquecimiento colectivo del proceso de aprendizaje sin límites geográficos. Actualización constante de los contenidos (deducción lógica del punto anterior). Reducción de costos (en la mayoría de los casos, a nivel metodológico y, siempre, en el aspecto logístico).

Es una alternativa de formación que no reemplaza necesariamente a los profesores y las clases presenciales, sino que es un espacio que desarrolla la autonomía del aprendiz.

2.1.2. *B-learning* o aprendizaje híbrido

En un concepto más relacionado con lo semipresencial, se encuentra el llamado *b-learning* (*blended learning*). Esto significa que un curso dictado en este formato incluirá tanto clases presenciales como actividades de *e-learning*[5].

Este modelo de formación hace uso de las ventajas de la formación 100 % en línea y la formación presencial, combinándolas en un solo tipo de formación que agiliza la labor tanto del formador como del alumno. La enseñanza combinada o mezclada, a veces también denominada enseñanza híbrida se define como cursos o programas en los que el contenido *online* supone entre un 30 % y un 70 % del total del curso. En contraste, la enseñanza presencial incluye aquellos cursos en los que el contenido *online* oscila entre 0 % y 29 % del curso. Las ventajas que se suelen atribuir a esta modalidad de aprendizaje son la unión de las dos modalidades que combina:

las ya comentadas que se atribuyen al *e-learning* y las de la formación presencial como: aplicación de los conocimientos e interacción física, lo cual tiene una incidencia notable en la motivación de los participantes, facilita el establecimiento de vínculos, y ofrece la posibilidad de realizar actividades algo más complicadas de realizar de manera puramente virtual.

2.1.3. Sistema de Gestión de Aprendizaje

Un sistema de gestión de contenidos es un programa que permite crear una estructura de soporte para la creación y administración de contenidos por parte de los participantes principalmente en páginas web. El entorno de *hardware* y *software* diseñado para automatizar y gestionar el desarrollo de actividades formativas se conoce como sistema de gestión de aprendizaje[18].

Este tipo de plataformas tecnológicas también se conoce como *Learning Management*

System. Sistema de Gestion de Aprendizaje (LMS). Un LMS registra usuarios, organiza catálogos de cursos, almacena datos de los usuarios y provee informes para la gestión. Suelen incluir también herramientas de comunicación al servicio de los participantes en los cursos.

Actualmente existe una gran oferta de plataformas, tanto de comerciales como de código abierto. En el ámbito universitario se está implantando con gran aceptación la plataforma de licencia libre Moodle[8] usada actualmente en la Universidad Simón Bolívar. En el ámbito comercial es muy famosa actualmente la plataforma Blackboard[2].

2.1.4. Modelo Vista Controlador

Modelo Vista Controlador (MVC) es un patrón de arquitectura de *software* para la implementación de interfaces de usuario. Divide la aplicación en tres partes interconectadas. Esto se hace con el fin de separar representaciones internas de la información con la forma en la que esta se presenta al usuario. Este patrón permite disociar los componentes promoviendo el reuso del código y el desarrollo en paralelo[20].

Tradicionalmente fue usado para la construcción de aplicaciones gráficas de escritorio, pero se ha vuelto popular para el diseño de aplicaciones web e incluso móviles. Los lenguajes de programación populares de la época poseen marcos de trabajo que facilitan la implantación de aplicaciones web usando este patrón.

- El Modelo: Es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación (lógica de negocio). Envía a la vista la parte de la información que en cada momento se le solicita para que sea mostrada.
- La Vista: Presenta la información en un formato adecuado para que el usuario pueda interactuar con ella.
- El Controlador: definido como la interfaz o intermediario entre la vista y el modelo. Su principal labor es la de responder a eventos e invocar peticiones al modelo cuando se hace alguna solicitud.

2.1.5. Arquitectura cliente-servidor

La arquitectura cliente-servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un sólo programa. existen distintos tipos específicos de servidores incluyen los servidores web, los servidores de bases de datos, los servidores del correo, etc. Mientras que sus propósitos varían de unos servicios a otros, la arquitectura básica seguirá siendo la misma[16].

2.1.6. Framework o entorno de trabajo

En el área de la computación un *framework* es una abstracción de *software* que provee una funcionalidad genérica que puede ser cambiada y adaptada con código escrito por el usuario para generar soluciones específicas. Provee facilidades y estándares para la creación y despliegue de aplicaciones en distintos nichos de la programación. Pueden incluir programas de soporte, compiladores, librerías de código, herramientas y *Application Programming Interface*. Interfaz de programación de la aplicación (API) que conjugan todos los componentes para permitir el desarrollo de un proyecto o sistema[14].

2.1.7. Manejador de bases de datos

Un manejador de bases de datos es un *software* que interactúa con los usuarios finales y otras aplicaciones para capturar y analizar datos. Un manejador de bases de datos de propósito general permite la definición, creación, consulta, actualización y administración de bases de datos. Existen varios famosos. Para este proyecto de pasantía se usó Microsoft *Structured Query Language*. Lenguaje de consulta estructurado (SQL) Server[21].

CAPÍTULO 3

MARCO TECNOLÓGICO

A continuación, se presentan los aspectos tecnológicos relacionados con el desarrollo del proyecto de pasantía, se describen brevemente y se indica su uso como parte de la solución propuesta.

3.1. Cliente

3.1.1. HTML

Hypertext Markup Language. Lenguaje de marcado de hipertexto (HTML) es un estándar para la creación de páginas web y aplicaciones web. junto con *Cascading Style Sheets*. Plantillas de estilos en cascada (CSS) y JavaScript forman las bases tecnológicas de la web[4].

HTML describe la estructura de la página web semánticamente e inicialmente le da una pista al navegador de como lucirá el contenido.

3.1.2. CSS

CSS son usadas para describir la presentación de un documento escrito usando un lenguaje de etiquetado, usualmente HTML.

CSS fue creado para una clara separación del contenido con la presentación, modificando el aspecto de las páginas, como tamaños, formas y colores de las distintas etiquetas[10].

3.1.3. JavaScript

Es un lenguaje de programación de alto nivel, débilmente tipado, multipropósito e interpretado. Es usado para darle interactividad a las páginas web, desde un sistema web hasta videojuegos. Es soportado por la mayoría de los navegadores actuales que implementan su propia representación de la especificación *ECMAScript*[11].

3.1.4. AJAX

Abreviación de *Asynchronous Javascript and Extensible Markup Language*. Lenguaje de marcado extensible (XML) es un conjunto de técnicas que hacen uso de varias tecnologías en el lado del cliente para crear aplicaciones web asíncronas. Con *Asynchronous JavaScript And XML*. JavaScript y XML asíncronos (AJAX), las aplicaciones web pueden hacer y recibir peticiones de datos sin interferir con la visualización y el comportamiento de la página existente, cambiando el contenido de la página dinámicamente sin tener que recargarla enteramente[9].

3.1.5. JQuery

Es una librería multiplataforma de Javascript diseñada para simplificar la escritura de programas que ejecuten uniformemente en los distintos navegadores. Es gratis amparado en la permisiva licencia del MIT. Además, permite la creación de módulos encima de Javascript que pueden ser compartidos para las funciones más básicas y comunes en la programación web[7].

3.1.6. Bootstrap

Es un *framework* de HTML, CSS y Javascript que facilita la construcción de páginas web que se adapten al ambiente donde son mostradas por medio de un sistema de cuadrícula con 12 columnas. Además provee algunos de los elementos usuales en las interfaces modernas como acordeones, paginación, botones desplegables que facilitan la construcción de interfaces web multiplataforma en distintos navegadores[3].

3.2. Servidor

3.2.1. PHP

Es un lenguaje de programación multipropósito mayormente usado para la programación web como lenguaje para la ejecución de tareas en el servidor. Originalmente creado por Rasmus Lerdorf en 1994 y ahora es producido por el Equipo de producción de PHP: *Hypertext Preprocessor* (PHP).

Es famoso por su amigable curva de aprendizaje, junto con la facilidad que ofrece para combinarse con el lenguaje de etiquetado HTML. El código es procesado por un intérprete de PHP implementado como un módulo en el servidor, que combina este resultado con el esqueleto de la página. Es altamente portable por ser distribuido como *software* libre bajo la licencia PHP y trabaja en todos los servidores web en casi todos los sistemas operativos[13].

3.2.2. Smarty

Smarty es un motor de maquetado para PHP. Específicamente, facilita la separación de la lógica de la aplicación y contenido de la presentación. Su mejor uso es descrito en la interacción de dos equipos de programación en el que uno genera las plantillas y el otro programa la aplicación, pero facilita la organización en desarrollos individuales. Haciendo usos de métodos como la inclusión y la herencia[12].

3.2.3. Microsoft SQL Server

Es un sistema de administración de bases de datos relacionales desarrollado por Microsoft usando como base el lenguaje *Structured Query Language*. Lenguaje de consulta estructurado (SQL). Tiene la función primaria de guardar y servir datos que sean requeridos por otras aplicaciones, que pueden correr en la misma computadora o en otra a través de una conexión en red[22].

3.2.4. Servidor HTTP Apache

Es un servidor web gratis y de código abierto ofrecido bajo los términos de la licencia Apache 2.0. Procesa las peticiones que llegan vía *Hypertext Transfer Protocol*. Protocolo de transferencia de Hipertexto (HTTP), el protocolo básico de la web. Apache es desarrollado y mantenido por una comunidad de desarrolladores pertenecientes a la Fundación de *Software* Apache. Es multiplataforma, funciona tanto en sistemas UNIX así como en Windows[6].

3.2.5. Swift mailer

Swift mailer es una librería de PHP que permite a cualquier aplicación escrita en este lenguaje el envío de correo electrónicos, proveyendo una interfaz flexible a través del paradigma orientado a objetos[17].

3.3. Pruebas

3.3.1. PHP Unit

Es un *framework* para la realización de pruebas de *software* basado en la arquitectura de creación de pruebas xUnit[1].

CAPÍTULO 4

MARCO METODOLÓGICO

En este capítulo se describe la metodología de trabajo utilizada en el desarrollo de este proyecto de pasantía.

Con el fin de enmarcar un desarrollo orientado a mejorar la productividad y calidad del *software*, que involucre además una reducción de riesgos y se adapte a las necesidades del cliente, se selecciona una metodología de desarrollo de *software* que moldea la construcción de características y funcionalidades a ofrecer por parte del *software* a través de prácticas de desarrollo que se adapten al mismo.

4.1. Naturaleza del proyecto

El trabajo realizado fue una extensión de un *software* existente. Una versión base de un Sistema de Gestión de Aprendizaje (SGA) que luego es instanciada para el uso de los distintos clientes de la compañía.

Las funcionalidades realizadas, en principio no tuvieron un cliente que especificara los requerimientos. Por lo tanto, los módulos se realizaron con la colaboración del equipo interno de la empresa que fungieron como dueños del producto y con la visión de agregar valor al sistema y equipararlo con otros sistemas del mercado.

Los requerimientos, al no estar fijados desde el inicio claramente, tenían la posibilidad de cambiar a lo largo del desarrollo. El proyecto fue dividido en pequeñas entregas funcionales para así obtener retroalimentación sobre la dirección que tomaba el proyecto.

4.2. Metodología ágil

La metodología Agile describe un grupo de principios para el desarrollo de *software* enmarcado en un ambiente en el que los requerimientos y las soluciones evolucionan a través del trabajo colaborativo entre los integrantes del equipo. Promueve planear adaptativamente, entregas tempranas, mejoramiento continuo, así como rápida y flexible respuesta al cambio.

Esta decisión fue tomada para aprovechar la flexibilidad que provee esta metodología. Este fue el elemento considerado como de mayor importancia dada la naturaleza del proyecto y se implementó en una de sus formas más comunes actualmente, *scrum*. A pesar de que *scrum* está planteado para ser usado por equipos de desarrollo fue ajustado para el proyecto en el que trabajo se realizó en su mayoría individualmente.

Esta metodología tiene un proceso de desarrollo iterativo incremental basado en entregas parciales y regulares del producto final al cliente, lo cual la hace flexible y de rápida adaptación ante cualquier cambio en cada iteración o Sprint. Dicha metodología está definida por los elementos descritos en las secciones siguientes de este capítulo.

4.3. *Scrum*

Scrum es un *framework* para la gestión del desarrollo de *software* aplicando la metodología ágil. Está diseñado para equipos de tres hasta nueve desarrolladores que dividen su trabajo en ciclos bisemanales llamados *sprints*, chequean su progreso constantemente y entregan *software* funcional al final de cada fase[15].

Cada miembro de un equipo *scrum* tiene especificado uno de los siguientes roles dentro del mismo:

4.3.1. Dueño del Producto o *Product owner*

Es aquel miembro del equipo que administra y define los requisitos del proyecto de desarrollo de *software*, así como sus objetivos, agregando y organizando estos requisitos de acuerdo a prioridades para “maximizar el valor del producto”. Asimismo, representa a todas las personas interesadas en los resultados del mismo.

Para este proyecto de pasantías, el gerente de ventas dentro de Fischer, Knoblauch & CO (FKC) asumió el papel de *Product Owner*.

4.3.2. Equipo

Equipo de profesionales autoorganizado, multidisciplinario y con un sistema jerárquico horizontal que desarrollan el proyecto. Preferiblemente, el equipo debe estar compuesto con un número suficientemente pequeño de miembros como para mantener las características

de “trabajo ágil”, pero lo suficientemente grande como para cumplir a tiempo todas las tareas. Este proyecto de pasantía fue realizado de manera estrictamente individual, por lo que el pasante asumió el papel de Equipo.

4.3.3. Facilitador o Scrum master

Es la persona encargada de liderar al equipo en miras de que todos los procesos internos se lleven de la mejor manera, cumpliendo con las reglas de *scrum*, a lo largo de todo el desarrollo del proyecto. Sirve de mediador entre el equipo de desarrollo y el dueño del producto, facilitando las reuniones y eliminando los impedimentos que puedan presentarse durante el desarrollo. Para este proyecto de pasantías, el jefe de la cuadrilla de programación de FKC asumió el papel de *Scrum* Master.

4.3.4. Stakeholders o Partes interesadas

Son aquellas personas para quienes el proyecto producirá el beneficio esperado que justifica su producción, pues son las interesadas en la realización del proyecto de desarrollo. Su participación se limita a las revisiones de cada sprint.

Miembros del equipo de programación y ventas formaban parte de las revisiones al final de cada sprint.

4.3.5. Eventos

Los eventos son todas aquellas reuniones planificadas para el seguimiento del proyecto de desarrollo y pueden ser:

4.3.6. Sprint

Es aquel período, de un tiempo previamente fijado y constante para todo el proyecto, durante el cual el equipo trabaja para convertir un subconjunto de requerimientos en una nueva versión del *software* totalmente operativo. Los sprints para este proyecto de pasantía tuvieron una duración aproximada de 10 días, por lo que se realizaron 10 sprints.

4.3.7. *Sprint Planning*

El *Sprint Planning* es una reunión que se realiza antes del inicio de cada sprint, donde el equipo de desarrollo determina la carga de trabajo que se compromete a completar en ese sprint, realizando la planificación del mismo. Las reuniones se dieron siempre con la presencia de el *Scrum master* y el dueño del producto.

4.3.8. *Daily Scrum*

Reunión diaria, de máximo quince minutos, en la que el equipo informa sobre el estado del proyecto. Cada miembro responde a las siguientes tres preguntas:

- ¿Qué hiciste ayer?
- ¿Qué harás hoy?
- ¿Has tenido algún impedimento para alcanzar tu objetivo?

Debido a que la pasantía se realizó de forma individual, esta reunión no se llevó a cabo, aunque fueron preguntas que se tuvieron en cuenta constantemente a lo largo del proyecto.

4.3.9. *Sprint Review*

Reunión que debe realizarse al final de cada sprint en la que el equipo de desarrollo presenta el trabajo completado durante el mismo a los interesados.

A los efectos de este proyecto de pasantía, esta reunión no se realizó de manera formal.

4.3.10. *Sprint Retrospective*

Después de cada sprint se lleva a cabo una retrospectiva del mismo, en la cual todos los miembros del equipo dan su opinión acerca del sprint recién superado en miras de mejorar continuamente el proceso de desarrollo. Como el proyecto de pasantía se realizó individualmente, esta reunión no se llevó a cabo.

4.3.11. Artefactos

La metodología *Scrum* hace uso de una serie de documentos que permiten su correcto funcionamiento y la comunicación del equipo completo. Entre ellos tenemos:

4.3.12. *Product Backlog*

Es un documento de alto nivel para todo el proyecto que consiste en una pila dinámica de requisitos denominados historias, descritos en un lenguaje no técnico y priorizados por valor de negocio. Decimos que es dinámica, pues los requisitos y prioridades se revisan y ajustan durante el curso del proyecto. Aquí, el *Product Owner* lista las características, funcionalidades, mejoras y correcciones del producto.

4.3.13. *Sprint Backlog*

Es un documento detallado y administrado por el equipo de desarrollo donde se describen, con una lista dinámica, todas las tareas a realizar para llevar a cabo las historias de un sprint.

4.4. Pruebas de *software*

Las pruebas del *software* no fueron requeridas por la empresa, por lo que no se les dio la atención necesaria. Pero al pertenecer al proceso de desarrollo de cualquier *software* de calidad se realizaron las que estuvieron al alcance de las limitaciones de tiempo. Entre estas estuvieron unitarias de cada módulo y de integración entre los módulos desarrollados, fue virtualmente imposible la creación de pruebas de integración con otros módulos del sistema debido a limitaciones de tiempo y a la ausencia total de pruebas de los demás módulos pertenecientes al sistema.

CAPÍTULO 5

FUNCIONALIDADES DESARROLLADAS DEL SISTEMA

En este capítulo se enuncian y describen las funcionalidades desarrolladas en el proyecto de pasantía, así como algunas previas que se relacionen o hayan sido modificadas para facilitar el entendimiento del lector.

Un Sistema de Gestión de Aprendizaje (SGA) se caracteriza por organizar todas las tareas que faciliten el entendimiento de un concepto por parte de los estudiantes. Entre estas existen cursos, chats, calendarios, estadísticas, etc.

Se debió a lo largo del proyecto implementar el soporte para un nuevo tipo de cursos: los cursos presenciales o seminarios. En los que un nuevo tipo de usuario: el instructor. Dicta una clase presencial sobre un tema que generalmente no puede ser cubierto mediante un curso en línea individual, debido a su complejidad, la importancia de la interacción en grupo para la realización de la actividad, entre otros.

Estos seminarios generalmente se realizan en ubicaciones físicas donde el instructor y los aprendices se encuentran cara a cara, pero también existe la modalidad de conferencias en línea para equipos internacionales.

Los usuarios podrían entonces escoger entre distintas sesiones en las que un seminario puede realizarse. Esto se realiza en respuesta a las limitaciones de tiempo que podría tener un grupo de estudiantes.

Las actividades previamente descritas son administradas por el usuario que toma el rol de administrador en el sistema, que tiene la potestad de crear, modificar y eliminar tanto los cursos como las ubicaciones.

Este relato surge del levantamiento de información realizado mayormente al principio del proyecto, en el que se identificaron las necesidades del cliente para los módulos y que se plasmó formalmente en el Diagrama de Casos de Uso que se muestra a continuación en la figura 5.1.

A continuación, se explican los componentes más importantes presentes en el Diagrama.

5.1. Actores

Para el desarrollo de los módulos se requirió de la participación de tres tipos de usuarios, administrador, instructor y aprendiz.

5.1.1. Administrador

El usuario administrador se encarga de tres tareas importantes según se visualiza en la figura 5.1. Gestionar los seminarios, las sesiones, las ubicaciones donde éstas se realizan y las notificaciones recibidas por los distintos tipos de usuarios.

5.1.1.1. Gestionar seminarios

Los administradores podrían listar los seminarios, crear un seminario nuevo y asignarlo a grupos de aprendices preestablecidos y por último pueden modificar los datos del mismo.

5.1.1.2. Gestionar ubicaciones

Un módulo de gestión de ubicaciones fue creado para su mejor visualización por medio de un mapa y reuso entre distintas sesiones. El administrador puede crear, editar, listar y modificar las ubicaciones que luego serán asignadas a las distintas sesiones. Además, puede especificar la capacidad de las mismas, así como visualizar en un calendario las sesiones planificadas para la misma.

5.1.1.3. Gestionar sesiones

Cada seminario, puede contener una cantidad arbitraria de sesiones, las funcionalidades son parecidas a las que posee sobre los seminarios con la diferencia de que para crear una sesión el administrador asigna un usuario del sistema como instructor, una ubicación preestablecida y fechas de inicio y fin en las que la sesión se realizará.

Las sesiones pueden ser tanto encuentros físicos como conferencias en línea, cuya diferencia es la asignación del tipo de ubicación.

5.1.1.4. Notificar usuarios

Los administradores tienen también la potestad de decidir si las notificaciones de interacción entre los aprendices y los instructores son enviadas o no. Como lo son las notificaciones de confirmación o cancelación por parte de los aprendices, o la cancelación de una sesión no vacía.

5.1.2. Instructor

El usuario instructor es el que se encarga del dictado de una lección, para esto necesita gestionar sus sesiones y manejar las calificaciones de los usuarios. Esta es una de las funciones que puede realizar el administrador a través de la gestión de sesiones.

5.1.2.1. Listar sesiones

Un administrador puede listar todas las sesiones de todos los cursos, un instructor solo podrá acceder a las sesiones de las que él sea instructor, valga la redundancia.

5.1.2.2. Imprimir lista de estudiantes

El instructor puede imprimir una lista de los usuarios confirmados para la sesión pertinente, con los datos importantes de la misma.

5.1.2.3. Evaluar estudiantes

Dentro del listado de los estudiantes confirmados para una sesión el instructor puede modificar la asistencia, así como indicar la aprobación de un curso por parte de un aprendiz.

5.1.3. Aprendiz

Un usuario aprendiz interactúa con los cursos que le han sido asignados, visualiza las ubicaciones y tiene un canal de comunicación con los instructores de las distintas sesiones.

5.1.3.1. Gestionar reservas

Un usuario puede listar las sesiones de los distintos cursos que le han sido asignados. Al hacerlo tiene la posibilidad de confirmar su asistencia a la sesión que le parezca más conveniente viendo los instructores, fechas y ubicaciones de cada una. Puede también cancelar una sesión a la que haya confirmado asistencia.

Además, puede descargar los datos de la sesión y agregarlos directamente a su calendario de preferencia.

Si el usuario posee una sesión aprobada en un seminario se mostrarán los datos que describan dicha sesión.

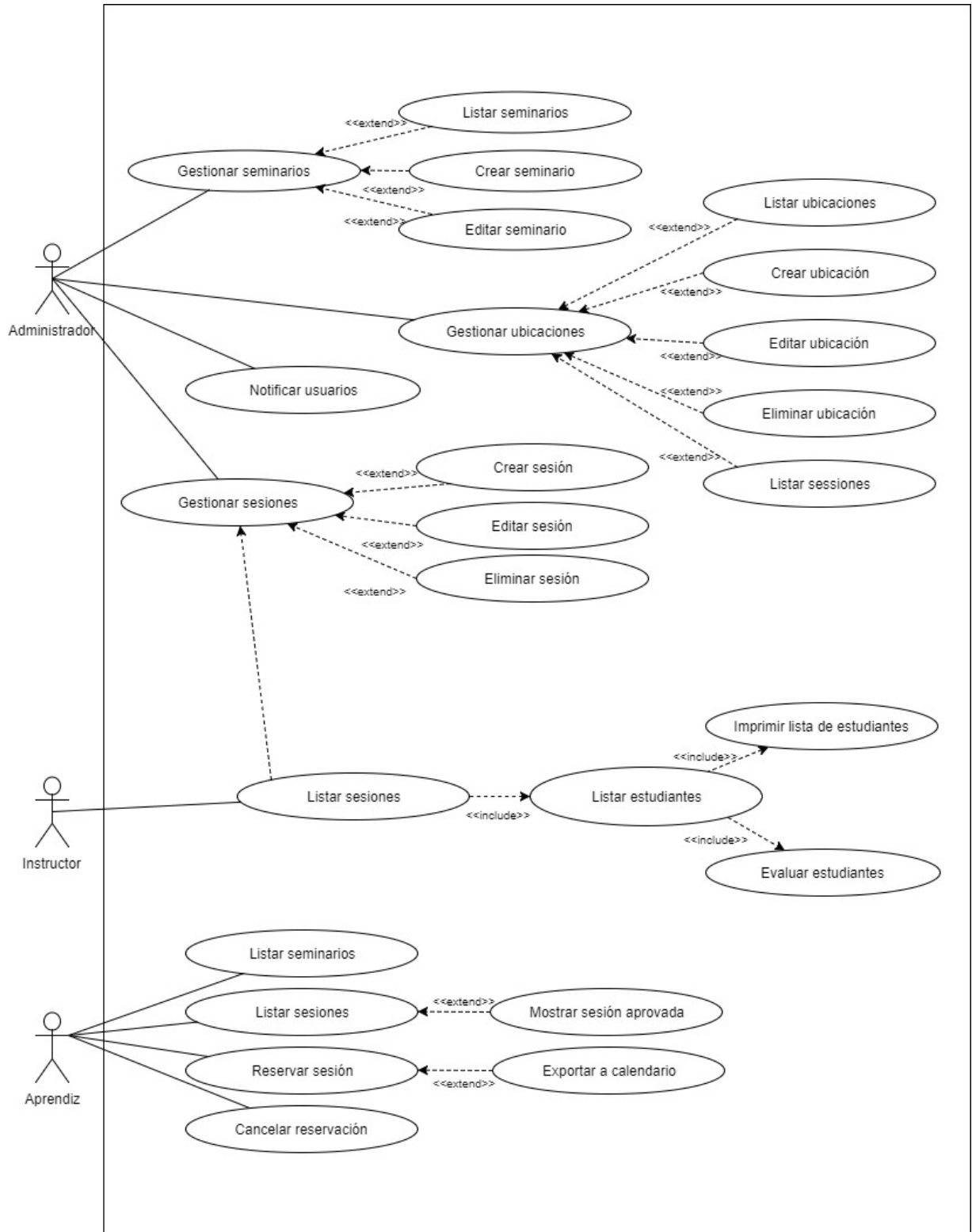


Figura 5.1: Diagrama de casos de uso.

CAPÍTULO 6

SOLUCIÓN DEL PROBLEMA

En este capítulo se expresan las técnicas utilizadas para el cumplimiento de las funcionalidades expresadas en el capítulo anterior.

El lenguaje de programación usado como base para el Sistema de Gestión de Aprendizaje (SGA) de Fischer, Knoblauch & CO (FKC) es PHP: *Hypertext Preprocessor* (PHP).

El trabajo se dividió en módulos. Se analizaron sus dependencias con el fin de crear un plan de trabajo organizado, que permitiera además la entrega constante de *software* funcional, apegándose al marco de trabajo iterativo.

A continuación, se expone una división de los retos más importantes y como fueron solucionados.

6.1. Módulo ubicaciones

Para soportar esta funcionalidad en la base de datos, se creó una tabla llamada ubicación con los datos que parecieron relevantes para el cliente, ilustrada en el Anexo A.3. Dicha ubicación estaría relacionada con las sesiones, pudiendo una ubicación alojar distintas sesiones. El SGA permite mantener distintos clientes para dar soporte de grandes compañías con filiales, por lo que las ubicaciones se construyen aisladas del grueso de la funcionalidad (los seminarios) para poder ser usadas luego en funcionalidades futuras o integrada en sistemas de los clientes activos de SGAs de FKC. Tienen una relación directa con la tabla cliente a través de una llave foránea. Para la clave primaria se usó un identificador creciente autogenerado.

Se acordó que el manejo de las ubicaciones se llevara a cabo en el área de administración de la aplicación, por el usuario administrador. Se agrego una nueva entrada de manejo de ubicaciones en la interfaz del administrador con este fin.

Se logró conseguir un producto gratis de esta aplicación que permitiera mostrar ubicaciones marcadas en un mapa generado por Google enganchado a un iframe en el sistema, con tan solo especificar medidas latitud y longitud, que tuvieron que ser agregadas luego a la tabla ubicación.

Para integrar la visualización de mapas usando la aplicación Google maps de manera gratuita para la empresa, el usuario debía ingresar los valores de latitud y longitud de la ubicación deseada. Para esto se ofreció como solución analizar gramaticalmente la *Unified Resource Locator*. Identificador de recursos unificado (URL) mostrada en la aplicación web google maps por medio de expresiones regulares para extraer los valores necesarios, facilitando así al usuario el proceso de agregar una ubicación sin que la empresa tuviera que usar las funcionalidades pagas de google.

6.2. Módulo Cursos del tipo Seminario

Luego de examinar las tablas y entender el funcionamiento de los cursos y como son asignados a los grupos (Anexo A.2) se decidió extender la tabla de curso como una generalización.

El sistema slo daba soporte a cursos del tipo multimedia dentro del mismo sistema, que podían ser de autoría del cliente o externos. Estos dos tipos eran manejados con un booleano “cursoDeAutoria” decisión de diseño que no daba espacio para la expansión de más tipos de cursos. Se sugirió agregar un campo “tipo” a la tabla de Curso asignándole el entero 2 al tipo de curso seminario. dejando los enteros 0 y 1 a los anteriores tipos de cursos. La migración de los tipos anteriores para ser representados con el nuevo campo “tipo” fue sugerida pero los demás desarrolladores rechazaron la propuesta por provocar cambios en otras funcionalidades del sistema. el estado final de la base de datos se demuestra en el Anexo A.3.

Una vez diferenciado el nuevo tipo curso de los demás en la aplicación se procedió a dar soporte a las actividades *Create, Read, Update, Delete*. Creación, Lectura, Actualización (CRUD) para los seminarios. para esto se reutilizó la sección del sistema usada para la creación de cursos. Agregando la opción *seminario* a las opciones previas en la creación de un curso y cambiando la estructura de la forma *Hypertext Markup Language*. Lenguaje de marcado de hipertexto (HTML) usando Javascript, así como los datos enviados mediante el método POST y su respectiva validación en el servidor para el soporte de los seminarios. La vista se incluyó en el Anexo B.3.

Una de las diferencias de los seminarios con los cursos multimedia es la forma en la que son aprobados, se decidió que podían existir dos tipos:

- Aprobar con tan solo asistir al curso.

- Aprobar asistiendo al curso y aprobando un examen presencial.

Para el soporte de estos dos tipos se agregó un campo “con_examen” que permite la representación.

Se aprovechó de los métodos de eliminación, listado y asignación a grupos que el sistema ya poseía, ajustándolos para el funcionamiento con los seminarios.

Para lograr esto debió separar visualmente los distintos tipos de cursos en las vistas de listado, para esto agrego un icono que los diferenciara en la vista de cursos del administrador (Anexo B.4). Además, agregó un filtro persistente en la sesión *Hypertext Transfer Protocol*. Protocolo de transferencia de Hipertexto (HTTP) que permitiera al administrador la visualización de solo cursos del tipo seminario a lo largo de su conexión.

En el método asignación se agregó luego un *preview* con las sesiones disponibles para el seminario visibles en el apéndice B.5 además del icono para los seminarios.

6.3. Usuario instructor

Para el manejo de la puntuación de las sesiones y el manejo de la asistencia se decidió crear un nuevo tipo de usuario en el sistema que pudiera apoyar al administrador en estas tareas. el usuario administrador tiene acceso a todas las funciones de un instructor en todos los cursos del sistema, mientras que el usuario solo tiene acceso a las sesiones en la que él mismo es el instructor.

Para esto se debió modificar el módulo de *login* para agregar la autenticación del nuevo usuario, que primero confirma si el usuario tiene una jerarquía más alta que la de instructor (un usuario de más rango que contenga sus funciones), en caso contrario, verifica si el usuario que intenta ingresar es instructor de algún curso.

El manejo de la autorización es realizado de forma manual, así que solo se debió poblar una estructura del tipo diccionario con las funcionalidades que se consideraran necesarias para el usuario instructor. En este caso, administrar los cursos de los que es instructor.

6.4. Sesiones de un seminario

Para soportar la funcionalidad en la base de datos se creó la tabla llamada Sesión que contiene todos los datos relevantes para la descripción de una sesión. Es importante recalcar que las sesiones sólo tienen sentido si forman parte de un curso del tipo seminario, cosa que se evidencia en la relación de agrupación en el modelo final de la base de datos en el Anexo A.3.

Un seminario puede estar compuesto por muchas sesiones, mientras que las sesiones no pueden ser compartidas entre los distintos seminarios, por lo que se representa como una relación 1 a muchos.

Una sesión se lleva a cabo en una ubicación por lo que se relacionan estas dos tablas. Una sesión puede suceder en solo una ubicación y las ubicaciones pueden albergar una cantidad de sesiones, por lo que la relación se modela como 1 a muchos.

Una sesión además es dictada/supervisada por un instructor por lo que se modela como una relación 1 a muchos análoga a las demás.

La última relación es la más interesante, las sesiones se relacionan directamente con los usuarios que asisten a ella (aprendices) y cada uno debe tener un espacio en la base de datos para modelar el resultado que este obtuvo en la misma. Es una segunda relación de la tabla sesiones con la tabla Usuario que en este caso es muchos a muchos pues un usuario puede participar en cualquier número de sesiones que le hayan sido asignadas, mientras que a una sesión concurren la cantidad de usuarios que su capacidad permita. Se modela con una tabla auxiliar que posee las dos foráneas y toda la información que la relación *sesión-usuario* necesitó al momento del desarrollo y la que pueda necesitar en el futuro, haciendo el diseño flexible, una de las metas del proyecto.

Se procedió a integrar el módulo con las ubicaciones. Para poder crear una sesión debía estar creada en primer lugar una ubicación, por lo tanto, se integró una entrada al módulo de ubicaciones desde el CRUD de la sesión que mantuviera en forma de *cookie* cual sesión estaba siendo modificada para volver a ella. La capacidad de la sesión no podrá ser mayor a la capacidad de la locación seleccionada.

Además, una sesión no podría ser creada en una fecha en la que la ubicación seleccionada estuviera ocupada (tuviera otra sesión cuyo horario choque). Esta limitación dio pie a que se implementara una visualización de las sesiones activas de una ubicación.

Para el campo de instructor podía ser seleccionado cualquier usuario activo del sistema del cliente en el que se esté realizando la operación. El cliente puede obtenerse por medio de la tabla curso, así que por normalización no se agregó información del cliente a la sesión como si se agregó a la ubicación.

Se agregaron campos que limitaban las confirmaciones y la cancelación de una sesión por parte de los aprendices, con el fin de dar seguridad al instructor de la cantidad asistentes a un curso.

Las sesiones tienen un campo “activa” para que puedan ser desactivadas en caso de cualquier eventualidad manteniendo los datos de la sesión para poder así ser reusados.

Las fechas de inicio y final de una sesión no pueden ser cambiadas si ya poseen usuarios confirmados, por lo que primero tendrían que ser eliminados y notificados de la misma para poder realizar los cambios.

El estado de un usuario con respecto a una sesión se modela en la tabla auxiliar Sesión-Usuario que puede apreciarse en el Anexo A.3. Donde además pueden agregarse más datos que describan esta relación dependiendo de necesidades futuras.

El borrado de una sesión se puede realizar desde el listado de las sesiones en el administrador como se muestra en las opciones del Anexo B.7. El usuario instructor no puede borrar la sesión, solo desactivarla.

Para la visualización se usó un *plug-in* de calendario bootstrap-year-calendar creado usando *bootstrap* y *jquery*. *Plug-in* que permite la representación de los meses en distintos lenguajes, así como la personalización de sus colores, aspectos necesarios para el amplio rango de necesidades de los clientes de FKC.

Este calendario se llena con *Asynchronous JavaScript And XML*. JavaScript y XML asíncronos (AJAX) a través de un *endpoint* creado para tal fin que devuelve las sesiones activas de la locación como un objeto *JavaScript Object Notation*. Notación de Objetos de JavaScript (JSON) que luego se muestra en los días del calendario como se muestra en el Anexo B.2. las sesiones desactivadas se muestran con un tipo de letra más pequeño y en cursiva para diferenciarlas de las demás.

6.5. Usuario aprendiz

El reto más importante de esta etapa del proyecto fue encontrar donde mostrar las sesiones de los cursos, si crear otra pestaña en el menú del sistema solo para los cursos del tipo seminario o integrar de alguna forma los cursos diferenciándolos de los demás.

La opción que se tomó fue la segunda. La primera opción agregaba una entrada más a un menú que comenzaba a lucir un poco abarrotado, pero, la segunda opción trajo sus dificultades. Se tuvo que modificar un código escrito de una forma complicada debido al mal uso de la librería de maquetado *Smarty*. La interfaz era una lista rígida de los cursos como se muestra en el Anexo B.11 por lo que se decidió dar una opción para generar una pestaña debajo de cada curso.

Este nuevo espacio generado debajo de los seminarios permitió colocar las sesiones disponibles del usuario, pero termino siendo un poco inconsistente que unos cursos tuvieran la opción de generar pestañas con más información y otros no. Como respuesta a esto los clientes sugirieron que ingresara un nuevo campo a la tabla curso para colocar una imagen que pudiera ser mostrada en todas las pestañas junto con la descripción del curso.

A simple vista era difícil para el usuario diferenciar los seminarios de los cursos multimedia por lo que se agregó un icono para los seminarios al lado del nombre de cada curso. Una vez visibles los seminarios, se consideró necesario notificar al usuario visualmente que la acción de confirmación de una sesión de seminario era necesaria o que dicha acción ya estaba realizada.

Se trabajó en conjunto al equipo de diseñadores gráficos para la realización de todos los iconos necesarios que mostraran el estado de un seminario, reservado, no reservado, sin sesiones disponibles como se muestra en el Anexo B.10.

Para el listado de las sesiones se hizo una consulta que buscaba en la base de datos las sesiones que estaban asignadas a los grupos a los que pertenecía el usuario y en cada caso también recolectaba las sesiones confirmadas del usuario para mostrar más información (Anexo B.13). Más adelante se desarrolló la calificación de la sesión y se consultó por las sesiones aprobadas del usuario para no mostrar las demás sesiones disponibles.

Se crearon dos *endpoints* para la confirmación y cancelación de las sesiones que se acceden mediante el ultimo botón en el listado de la sesión. si no hay una sesión confirmada por el usuario es posible confirmar cualquiera de ellas. Al poseer una sesión confirmada el usuario solo tiene la opción de cancelar la sesión confirmada actual y se hace el chequeo

en el *endpoint* de que la cancelación sea posible dados los datos de la sesión al igual que en la confirmación.

Al ser la mayoría de los clientes usuario del servicio de correo *outlook* fue solicitado por los clientes una facilidad para exportar automáticamente al calendario *outlook* del aprendiz. Para lograr esto se realizó una investigación sobre la especificación de formatos para la transmisión de eventos a través del protocolo HTTP *iCalendar* usado previamente por la empresa. *iCalendar* es soportado por las plataformas más famosas como *google*, *apple*, *microsoft outlook* entre otras. Se creó un pequeño módulo que recibía los datos de una sesión del sistema y producía un archivo en formato ics para ser descargado por el usuario y usado en su aplicación de calendario de preferencia. Con esto se pudo cumplir el requerimiento del cliente y más.

En la revisión continua de las funcionalidades realizadas surgió la necesidad por parte del cliente de enviar mensajes al instructor de una sesión directamente desde el listado de las sesiones disponibles. Para esto se realizó una conexión del listado de sesiones disponibles con el módulo de mensajería interna del sistema, colocando automáticamente como destinatario al instructor del curso.

6.6. Calificación de las sesiones

Un usuario solo debe poseer una única entrada en la tabla Sesión-Usuario que lo relacione inequívocamente con una sesión, para esto se utilizó como clave primaria de la tabla la combinación de las foráneas *sesion_id* y *usuario_id*. Además, se implementó una excepción para arrojar un error al usuario y notificar al administrador del sistema por medio de una entrada en la bitácora del sistema si este caso llegara a suceder.

Una vez asegurada la unicidad de la relación usuario cursa una sesión se puede proceder a calificarla. Tanto el administrador como el instructor pueden calificar una sesión. La calificación de la sesión permite al usuario aprendiz cambiar el estado del seminario que se encuentra en la tabla Curso. Para facilitar la explicación se construyó un diagrama de estados que demuestra los posibles cambios (Anexo A.1).

Para la calificación se construyeron dos *endpoints* uno para la modificación de la asistencia y otro que soporta la modificación del estado aprobado/reprobado. que solo pueden ser alcanzados si la fecha de inicio de la sesión ha sido alcanzada.

El proceso puede resumirse en:

1. Un administrador asigna un seminario a un grupo compuesto por una cantidad de usuarios (Anexo B.5).
2. Una vez asignado el seminario, todas las sesiones que lo componen son asignadas como posibles para los usuarios del grupo a través de la interfaz de listar cursos del Anexo B.10.
3. El usuario puede confirmar cualquier sesión que no esté llena.
4. Con una sesión confirmada el usuario puede cancelarla y volver al punto 1 para elegir una sesión distinta.
5. El usuario que confirmó una sesión asiste a ella. El instructor por lo tanto puede marcar la asistencia en la interfaz construida para ello que se muestra en el Anexo B.8. Si el seminario no posee examen final, la sesión y por lo tanto el seminario, son aprobados.
6. Si el seminario tiene un examen presencial se le presenta la opción de notificar que el aprendiz aprobó el examen, en caso de dejar este campo en blanco se intuye que el usuario reprobó dicho examen.
7. En el caso de que el aprendiz sea reprobado el seminario se marca como reprobado hasta que el usuario confirme otra sesión.

Se creo una planilla de estilos *Cascading Style Sheets*. Plantillas de estilos en cascada (CSS) distinta para la vista de impresión de la página de la lista de aprendices de una sesión y que esta pudiera guardarse como un *Portable Document Format*. Formato de documento portable (PDF).

6.7. Sesiones de tipo *online* para los seminarios

Añadir sesiones en línea fue un proceso parecido al de agregar los seminarios a los cursos. Se especificaron los datos que diferenciaban a las sesiones en línea de las ya existentes y se decidió por lo tanto crear una especialización de la tabla Sesión en física y “en línea” (Anexo A.3).

Se ingresó un campo tipo a la tabla Sesión para diferenciarlas. el entero cero representa las sesiones físicas y el uno se refiere a las en línea.

Para crear una sesión online se agregó una opción de tipo en la creación de las sesiones (Anexo B.6) al igual que con los cursos.

Este tipo de sesiones debían ser mostradas de forma distinta en las vistas que involucraran las sesiones ya que estas cuentan con datos distintos. por lo que se modificó la visualización del antiguo modal del aprendiz para la ubicación. En el caso de una sesión física se seguiría mostrando el mismo modal y en caso contrario se mostraría el modal con los datos referentes a la sesión en línea solo en el caso en que el usuario haya confirmado dicha sesión.

En lugares donde se listan las sesiones el campo de ubicación paso a llamarse “ubicación/url” y en el caso de una sesión en línea se muestra e texto “sesión en línea” en vez del nombre de la ubicación.

6.8. Notificaciones

Las notificaciones en el sistema generalmente son hechas a través de correos, se decidió mantener esta práctica. Se utilizó para este propósito la librería *swift mailer*.

El proceso de agregar las notificaciones paso por crear una función de envoltura a la librerías de *swift mailer* que recibiera los datos básicos necesarios para enviar un correo como: encabezado, mensaje y destinatarios dentro de un arreglo.

El resto fue ubicar los envíos de correos en los sitios adecuados, llenando los datos correctos según fuera el caso.

Al comenzar a probar el sistema, los clientes se dieron cuenta que las notificaciones podían llegar a ser demasiadas en cursos compuestos por muchos estudiantes, por lo que luego se desarrolló una opción para desactivar los correos recibidos por el instructor de una sesión. como ejemplo se muestra uno de los correos enviados por la aplicación en el Anexo B.14.

6.9. Estadísticas

Se debió modificar algunas consultas en las estadísticas para que estas pudieran reflejar los resultados de los seminarios. generalmente suavizando condiciones de los *joins*. Se diferenciaron los seminarios con el uso de un icono distintivo.

Además, algunos datos en la tabla de visualización de las estadísticas no concordaban con los datos referentes a los seminarios. Estos datos se omitieron como: momento de inicio, momento de terminación, tiempo de aprendizaje que más adecuados y fácilmente medibles en los cursos web.

6.10. Adaptación de los módulos al SGA de Bibliomed

El SGA de Bibliomed poseía una abstracción por encima de los cursos llamada Módulos, que hizo que la integración no fuera tan trivial como era esperado. Además, el cliente agregó nuevas condiciones a la interacción de los seminarios.

Los módulos no son más que agrupaciones de cursos en línea que deben ser realizados para su finalización. El cliente especificó que los seminarios podían ser un reemplazo o un correquisito a los cursos en línea, es decir, según los requerimientos del creador del módulo las formas de aprobarlo serían:

1. Aprobar todos los cursos en línea.
2. Aprobar el seminario asociado al Módulo.
3. Aprobar ambos, cursos en línea y el seminario.

La relación entre módulos y seminarios es 1 a 1, es decir, solo puede existir un seminario dentro de cada módulo, de igual manera el esquema se realizó con la flexibilidad necesaria para permitir más seminarios por módulo en el futuro. Para esto se colocó la foránea identificadora del módulo en la tabla Seminario y no en los dos como es comúnmente implementado. Que un seminario no pueda ser asignado a dos módulos distintos se asegura a través de la interfaz. Un cambio a una interfaz más permisiva puede hacer que se necesite especificar la foránea de los dos lados de la relación.

Se agregó también el campo “modo_aprobacion” para representar los modos de aprobación descritos anteriormente.

El resto del esquema termina siendo el mismo que en el SGA base de FKC. uno de los objetivos de la integración fue alejarse lo menos posible del sistema base para facilitar la mantenibilidad. Por lo que los procesos de integración fueron realizados, pero son análogos a los anteriormente descritos.

En el SGA de Bibliomed la creación de un curso del tipo seminario paso a ser la asignación de un seminario a un módulo. Como los seminarios solo pueden existir asignados a un módulo, la asignación se hace desde la vista de edición del módulo como se demuestra en el Anexo B.16 en contraste con la creación de curso del tipo seminario del Anexo B.3.

Al realizarse desde la vista de actualización del módulo se asegura que el seminario exista solo en ese módulo, ya que no puede elegirse un seminario previamente creado.

Se integraron las sesiones confirmadas por el usuario con un módulo de calendario para el usuario implementado en el SGA de Bibliomed. Agregó a las consultas previas que llenaban el calendario una consulta que ofreciera las consultas confirmadas y atendidas por el usuario de manera que se mostraran en su calendario personal. Cuidando mantener la misma estructura de los eventos para su correcta presentación.

CAPÍTULO 7

DESARROLLO DE LAS FUNCIONALIDADES

En este capítulo se describe el orden en el que las tareas fueron realizadas durante el desarrollo del proyecto de pasantía. Bajo las directrices de la metodología *scrum* y a lo largo de diez sprints, comprendiendo las fases: especificación y análisis de requerimientos, diseño e implementación, e implantación de los cambios realizados a Sistema de Gestión de Aprendizaje (SGA) de Fischer, Knoblauch & CO (FKC). A continuación, se describen las actividades realizadas en cada fase, así como actividades fuera de la cobertura del plan de trabajo que se realizaron para FKC.

7.1. Primer Sprint

7.1.1. Objetivos

- Familiarizarse con el ambiente de trabajo de la empresa.
- Aprender a usar el lenguaje de programación PHP y sus buenas prácticas.
- Analizar a fondo el funcionamiento del SGA a extender.
- Levantamiento de requerimientos del proyecto a realizar.

7.1.2. Actividades

- Familiarización con las herramientas

El pasante no poseía experiencia previa con el lenguaje de programación usado en la empresa, PHP, por lo que se acordó la exploración de referencias sobre el funcionamiento y el correcto uso de dicho lenguaje.

Se usaron distintos recursos tanto literarios como web, mayormente la página web que contiene la documentación oficial del lenguaje como referencia. [13]

- Análisis a fondo el funcionamiento del SGA

Para esto el pasante tuvo que instalar las herramientas comunes de desarrollo en inglés, puesto que recibió un ambiente completamente en alemán. Entre estos: siste-

ma operativo, manejador de las distintas bases de datos Microsoft SQL y Microsoft Access, y el navegador.

Una vez instalado el ambiente de desarrollo adecuado el pasante procedió a explorar el sistema. Rápidamente se dio cuenta que el código fuente escrito estaba muy desorganizado. Código alto acoplamiento en el que se mezclaban lógica del negocio con la presentación. constante uso de instrucciones *Structured Query Language*. Lenguaje de consulta estructurado (SQL) construidas dentro de cada vista susceptibles a inyecciones de SQL. Muy bajo reúso de código a lo largo de la aplicación y técnicas de programación desactualizadas para el código PHP escrito en la actualidad especialmente al momento de recuperar información de la base de datos. El código fuente no describía ninguno de los patrones de diseño que podían ayudar para la construcción de sistemas de este tipo, como composición, observador, entre otros. No existía para el sistema en cuestión ningún tipo de pruebas, ni documentación que apoyara al pasante en esta exploración.

Se descubrió el uso del lenguaje de maquetado Smarty que permite la separación de la capa lógica y la de presentación y se procedió a conseguir referencias para el aprendizaje de esta librería.

Se estudió además el esquema de la base de datos usando la herramienta *SQL Management Studio* que genera automáticamente un esquema visual de la base datos, donde se buscó entender los patrones con los que fue construida con el fin de mantener consistencia en las nuevas funcionalidades a desarrollar. Entre estas, implementación de las relaciones entre tablas, nombramiento de los campos, así como el tipo y tamaño de los mismos.

Asimismo, se analizó la estructura de los archivos, para mantener la misma estructura con la que estaban ordenados, separando los distintos componentes de la aplicación como archivos de código PHP, Javascript, *Cascading Style Sheets*. Plantillas de estilos en cascada (CSS) y archivos estáticos. Se evidenció una estructura en el nombramiento de los archivos que se siguió a lo largo del desarrollo, colocando primero el nombre de lo que podría llamarse módulo y luego la acción específica dentro del mismo, por ejemplo: `seminar_session_create`, `seminar_session_update`, `location_create`, etc.

- Levantamiento de requerimientos

Al terminar el análisis de la base de código y entender a grandes rasgos su funcionamiento y estructura se procedió a hacer el levantamiento de los requerimientos necesarios para la extensión. El objetivo era dividir el proyecto en piezas de funcionalidad con el fin de obtener una visión más clara y objetiva de las necesidades del

cliente, así como un mapa que permitiera al pasante crear un plan y una estimación para la realización del proyecto. De esta reunión surgió el diagrama de casos de uso (anexo 5.1.)

- Exploración de otras plataformas

En esta fase también se realizó una investigación sobre la implementación de esta funcionalidad en otros SGA como e-front y moodle con el fin de tener una referencia de un producto que ya se encuentra en el mercado.

7.2. Segundo sprint

7.2.1. Objetivos

- Desarrollo del módulo ubicaciones que sirvan como locación de los seminarios.

En este sprint se desarrolló el módulo de manejo de ubicaciones representado en el diagrama de casos de uso (Anexo 5.1). Se decidió iniciar con este módulo por ser una funcionalidad aislada, sencilla y componente necesario para la creación de sesiones presenciales. Tomando así un acercamiento de abajo hacia arriba en el desarrollo del proyecto.

7.2.2. Actividades

- Ampliación de la base de datos
- Creación del *Create, Read, Update, Delete*. Creación, Lectura, Actualización (CRUD)

Se procedió entonces a la construcción de interfaces que permitieran la creación de una ubicación nueva, listar las ubicaciones existentes, modificar una ubicación existente tanto como eliminar una ubicación. Se hizo énfasis en que las interfaces creadas siguieran un aspecto consistente con las otras funciones de administración ilustradas en los apéndices B.1 y B.2.

- Integración con google maps

Al terminar la funcionalidad básica el dueño del producto sugirió integrar las ubicaciones del sistema con la aplicación Google maps. Tarea para la cual el pasante debió investigar sobre el uso de este *Application Programming Interface*. Interfaz de programación de la aplicación (API).

Se recomienda en un futuro el pago de este API (Javascript de google maps) si se desea facilitar aún más la interacción con mapas dentro de la aplicación. Con el beneficio de no depender que en algún momento el proveedor google modifique la estructura de sus *Unified Resource Locator*. Identificador de recursos unificado (URL) o desaparezca los datos de latitud y longitud de las mismas.

En el Anexo B.2 se muestra como se integró el mapa a la vista de editar ubicación.

7.3. Tercer sprint

7.3.1. Objetivos

- Analizar la estructura de cursos previamente implementados en el sistema.
- Desarrollo e integración del módulo cursos del tipo seminario.
- Crear un nuevo usuario para el sistema gestionar solo cursos presenciales.

7.3.2. Actividades

- Análisis de los cursos implementados en el sistema
- Desarrollo del módulo cursos del tipo seminario
- Creación del usuario instructor

7.4. Cuarto sprint

7.4.1. Objetivos

- Desarrollar el módulo de sesiones de seminario.
- Integrar el nuevo módulo con los demás módulos del sistema.
- Agregar visualización de sesiones activas para una ubicación.

En este sprint se desarrolló el módulo de gestión de las sesiones de un seminario representado en el diagrama de casos de uso (Anexo 5.1), la parte más importante del proyecto.

Interactúa con los demás módulos implementados y es la funcionalidad central que da vida al sistema. Es la información más importante que se guarda en la base de datos, Los datos de las sesiones y la interacción de los usuarios con éstas.

Cabe destacar que al momento de mencionar sesión se hace referencia a las sesiones de los seminarios y no la sesión *Hypertext Transfer Protocol*. Protocolo de transferencia de Hipertexto (HTTP) a menos que se especifique lo contrario.

7.4.2. Actividades

- Representación en la base de datos
- Creación del CRUD

El proceso básico de creación del CRUD fue muy parecido al CRUD de las ubicaciones, pero con más consideraciones a tener en cuenta, pues aquí se integraban distintas partes de la aplicación, como un instructor, que debía ser un usuario del sistema; una ubicación, en la que se llevaría a cabo la sesión creada, manejo de fechas, activación y desactivación de las sesiones. La vista de CRUD se presenta en el Anexo B.6.

- Creación de una sesión
- Actualización de una sesión
- Borrado de una sesión

El borrado básico se realizó en este sprint, pero luego tuvo restricciones que surgieron del manejo de las calificaciones implementado en el sexto sprint y además se agregaron opciones de notificación a los usuarios de una sesión cancelada en el sprint siete.

- Visualización de sesiones activas de una ubicación

Se agregó para facilitar el proceso de elegir una locación libre en la fecha de la sesión.

7.5. Quinto sprint

7.5.1. Objetivos

- Diseño de la interfaz para la gestión de sesiones.

- Desarrollar las funcionalidades para el aprendiz.

En este sprint luego de haber realizado los requerimientos básicos necesarios para el soporte del módulo o lo que generalmente es llamado *backend* de la aplicación. Se procedió a realizar el *frontend* que permitiera a los usuarios interactuar con los cursos del tipo seminario y sus sesiones. Específicamente los casos de uso pertenecientes al usuario aprendiz en la parte baja del diagrama de casos de uso (Anexo 5.1).

7.5.2. Actividades

- Diseño de la interfaz
- Listado de las sesiones disponibles
- Confirmación y cancelación de una sesión
- Exportar sesión al calendario
- Integración con el módulo de mensajería interna del sistema
- Modal con los datos de la ubicación

Se integró una ventana modal para mostrar directamente el mapa y los datos de la ubicación que eran previamente mostrados en el módulo de ubicación. Usando el mapa el usuario puede guardar la ubicación directamente en su cuenta *google* si se encuentra autenticado con esta aplicación.

7.6. Sexto sprint

7.6.1. Objetivos

- Implementar las funcionalidades de calificación de las sesiones de seminarios.
- Generar *Portable Document Format*. Formato de documento portable (PDF) con la lista de estudiantes de una sesión.
- Implementar funcionalidad para hacer el módulo de seminarios opcional.

En este sprint, una vez que se contó con confirmaciones válidas que procedían desde usuarios aprendices del sistema se procedió a implementar la calificación por parte del usuario instructor y administrador.

7.6.2. Actividades

- Calificación una sesión
- Actualizar la funcionalidad de borrado de una sesión

Se decidió agregar condiciones para el borrado de las sesiones al desarrollar esta funcionalidad. Para borrar una sesión esta no debe tener usuarios que la hayan confirmado. Tampoco debe tener resultados de ningún usuario. Borrar una sesión con resultados eliminaría también el estado de aprobado que pudieran tener algunos usuarios.

- Generación de PDF

Se permitió además al usuario instructor la opción de imprimir una lista con los estudiantes de la sesión que contiene los datos de la sesión y los estados posibles de cada uno dependiendo del tipo de seminario. El resultado se muestra en el Anexo B.9.

- Módulo de seminarios como una opción

Dado que el SGA es un producto que se personaliza para los distintos clientes de la compañía, los clientes se interesaron en tener la opción de crear una nuevo SGA sin la funcionalidad del módulo seminario creado por el pasante. Para poder ofrecerlo con una opción extra que pudiera flexibilizar el esquema de precios del producto.

Para esto el pasante debió modificar la funcionalidad de otro tipo de usuario: *super admin*. Específicamente la creación y actualización de un cliente. Dependiendo de si los cursos del tipo seminario estaban activados o no, todas las opciones relacionadas con este módulo debían desaparecer. Entre estas:

- Creación de un curso del tipo seminario.
- Manejo de las ubicaciones si éstas solo son usadas para sesiones de seminarios.
- Visualización de los cursos del tipo seminario y sus sesiones a lo largo de la aplicación.
- Desactivación del nuevo tipo de usuario creado por el pasante *instructor*.

7.7. Séptimo sprint

7.7.1. Objetivos

- Crear nuevo tipo de sesiones: *sesiones en línea*.
- Implementar notificaciones para los eventos del módulo.
- Integrar los cursos del tipo seminario a las estadísticas del sistema.

A la mitad del desarrollo del sprint anterior el cliente consideró agregar una nueva funcionalidad al módulo. La posibilidad de agregar sesiones dentro de los seminarios que pudieron darse en conferencias en línea. El pasante decidió incluir esta funcionalidad en el presente sprint que ya contenía el envío de notificaciones por correo en los principales eventos del sistema y la integración de los seminarios en las estadísticas de los cursos.

Con este sprint se finalizó el desarrollo del módulo de seminarios excluyendo futuras correcciones.

7.7.2. Actividades

- Sesiones en línea en la base de datos

La diferencia entre estas sesiones y las anteriores es que estas ya no tendrían asignadas una ubicación física, en este caso tendrían una URL y un código de acceso para ingresar a la misma.

- Presentación de las sesiones en línea

La lógica de negocio detrás de las sesiones en línea terminó siendo la misma que para las sesiones físicas por lo que la calificación de éstas no debió ser modificada. El reto estuvo en la creación, actualización y presentación de las mismas al usuario.

- Notificaciones

Se procedió primero a identificar cuáles eran los eventos que requerían de notificaciones a los usuarios entre estos se encontraron:

- Información de una sesión confirmada o cancelada por el usuario.
- Correo informando al usuario que una de sus sesiones confirmadas fue cancelada.

- Notificar al instructor que un usuario ha confirmado o cancelado una de sus sesiones.

- Integración de las estadísticas

Las estadísticas es una de las partes más importantes en el manejo de los cursos (Anexo B.15). Allí los administradores pueden corroborar el desempeño de los aprendices. Al ser los seminarios un nuevo tipo de curso también debían ser incluidos en las estadísticas. La empresa proveyó un mapa de los estados de los cursos al que el pasante tuvo que moldear los resultados de las sesiones.

7.8. Octavo sprint

7.8.1. Objetivos

- Integrar el módulo de seminarios al SGA de Bibliomed.

A mitad del desarrollo del módulo uno de los clientes de FKC (Bibliomed) poseedor de su SGA se interesó por el módulo de seminarios. Por lo que la primera tarea al finalizar el desarrollo fue integrar las funcionalidades al sistema personalizado para esta compañía.

Se decidió entonces hacer las modificaciones necesarias para el soporte de estas peticiones en la base de datos del SGA de Bibliomed. El resultado final quedó plasmado en el Anexo A.4.

7.8.2. Actividades

- Adaptación de la base de datos
- Asignación de un seminario
- Modificación de la interfaz del aprendiz

La interfaz del listado de los módulos disponibles para el aprendiz se adaptó para el soporte de los seminarios la interfaz anterior se muestra en el Anexo B.17 y la interfaz final en el Anexo B.18.

- Aprobación de los módulos

El mayor desafío de este sprint fue la programación de la aprobación de los módulos. Para esto el pasante trabajo muy de cerca con el desarrollador del SGA de Bibliomed, asegurandose que se modificaba correctamente el algoritmo para la aprobación de un módulo de aprendizaje.

El pasante proveyó una interfaz para la fácil consulta de terminación de un seminario dentro de un módulo y juntos unieron las culminaciones de los dos tipos de cursos para cumplir las restricciones pedidas por el cliente para la aprobación de los módulos.

- Integración con el calendario del usuario

7.9. Noveno sprint

7.9.1. Objetivos

- Migrar el SGA base de FKC para el manejo de caracteres UTF-8.
- Implantar el SGA de FKC en los ambientes Munich y producción.
- Implantar el SGA de Bibliomed en el ambiente de producción.

Hasta el momento todo el desarrollo se había realizado en servidores de prueba, en este sprint se traslada la base de código actualizada a los servidores de producción de cada empresa y se realizan las pruebas correspondientes.

7.9.2. Actividades

- Migración para el soporte de caracteres UTF-8

Al integrar las nuevas funcionalidades en el SGA de Bibliomed el pasante se dio cuenta que muchos de los caracteres especiales de lenguaje alemán perdían representación. Esto se debió a que el SGA de Bibliomed representaba caracteres en el formato UTF-8 en comparación con el más viejo ISO del sistema base.

Bibliomed al soportar el lenguaje ruso usa la mejor opción UTF-8. Pero mantener este esquema significaría que los próximos sistemas vendidos por FKC también

podrían luego necesitar la migración a UTF-8 para soportar otros lenguajes. El equipo decidió entonces asignar la tarea al pasante de migrar el sistema al soporte de caracteres UTF-8 en su versión básica.

Para lograr esto debían guardarse todos los archivos con el nuevo formato que previamente estaban guardados en el formato ISO y cambiar la etiqueta *Hypertext Markup Language*. Lenguaje de marcado de hipertexto (HTML) que especifica el *charset* en todos los archivos del sistema. Se utilizaron expresiones regulares dentro del editor de texto para modificar este parámetro satisfactoriamente.

Los otros desarrolladores realizaron las pruebas pertinentes para comprobar el resultado de la operación.

- **Implantación del SGA de FKC**

El proceso de implantación del sistema consistió en modificar las tablas en el servidor objetivo y enviar los archivos a través de *File Transfer Protocol*. Protocolo de transferencia de archivos (FTP). Este era un proceso totalmente manual.

Para FKC debió hacerse en dos servidores distintos, uno para la filial en Munich y el otro para producción.

Se realizaron pruebas manuales para la integración del módulo con las demás partes del sistema.

- **Implantación del SGA Bibliomed**

El proceso fue análogo a la implantación del SGA de FKC solo cuidando las diferencias en la estructura de la base de datos.

7.10. Décimo sprint

7.10.1. Objetivos

- Explicar a los demás desarrolladores la estructura de la solución implementada.
- Actualizar los esquemas de la base de datos de la empresa.
- Realizar del informe de pasantía.

Durante este sprint el pasante mantuvo charlas con los desarrolladores sobre el desarrollo del módulo, explicando cuales partes podían ser mejoradas, describiendo los *plugins* utilizados, respondiendo preguntas y haciendo correcciones menores al sistema.

El pasante expuso sus recomendaciones para el mejoramiento del sistema y los métodos de trabajo, así como también recibió sugerencias sobre el trabajo realizado.

7.10.2. Actividades

- Actualizar los esquemas de la base de datos de la empresa

Se actualizaron los esquemas generados en la herramienta SQL Management Studio para que mostraran los cambios hechos por el pasante en los sistemas de FKC y Bibliomed.

- Realización del informe de pasantías

El último sprint fue mayormente usado para la realización del informe de pasantías a entregar en la Universidad Simón Bolívar.

7.11. Actividades extra

Durante la realización del proyecto dada la naturaleza de la empresa como consultora informática de aprendizaje electrónico el pasante fue requerido para la realización de tareas extra de los diversos clientes de FKC.

7.11.1. Exportar preguntas de un SGA

Una funcionalidad para un sistema realizado con *Active Server Pages* (ASP). Consintió en la implementación de una migración para los datos de uno de los clientes. Específicamente, las preguntas presentes en su base de datos. El pasante debió automatizar la generación del nuevo formato para las preguntas, marcando las distintas preguntas con sus opciones con un carácter especial marcando la respuesta correcta y separando las preguntas según su tipo (selección simple, compuesta). Luego sirviendo el resultado a través de la web en el sistema del cliente en un archivo que pudiera ser interpretado en las *suites* ofimáticas comunes.

Para la realización de esta tarea el pasante realizó la lógica enteramente dentro del lenguaje SQL, Limitando el uso de ASP al servicio de los resultados a través de la página web.

7.11.2. Migración a UTF-8

La empresa al estar satisfecha con la migración de su SGA base a la codificación *Unicode Transformation Format 8-Bit*. Formato de transformación de 8 *bits* (UTF-8) requirió que el pasante migrara un SGA de un cliente a esta codificación para el soporte del lenguaje turco.

7.11.3. Funcionalidad en modulo *responsive*

El pasante apoyó a otro desarrollador en uno de sus proyectos, una herramienta para generar cursos o presentaciones dentro de uno de los SGA de un cliente.

La herramienta permite la construcción de diapositivas con distintos elementos de interfaz básicos, como, imágenes con texto, figuras, etc. El pasante codificó los parámetros de entrada y la presentación de imágenes con leyendas al pie o encabezando las mismas, enmarcado dentro del diseño *responsive*.

7.11.4. Imágenes de las vistas de login

Los SGA vendidos por FKC son personalizables, entre uno de los parámetros que pueden modificarse para la adaptación a los clientes son las imágenes de la vista de inicio de sesión, y el *banner* que se muestra al tope de cada vista del sistema tanto en la vista de usuario como en la vista de administrador.

Anteriormente, los desarrolladores dentro de FKC recibían las imágenes necesarias para estas vistas y las adaptaban al sistema de manera manual y colocándolas en un fichero creado para este propósito en el servidor.

El pasante agregó una funcionalidad para agregar estas imágenes y su color de fondo directamente desde la interfaz de super admin dentro del sistema que soporta la previusualización de la imagen escogida. Ofreciendo una manera más rápida y sencilla para la elección de dicha imagen, así como sus dimensiones y además permitiendo que los mismos clientes puedan modificar sus imágenes de inicio de sesión y *banner*.

7.12. Dificultades encontradas

A lo largo de la realización del proyecto de pasantía se encontraron las siguientes dificultades:

- El idioma fue una limitante para el pasante a la hora de hacer el análisis del sistema y del código a modificar en algunos casos. El código muchas veces no explicaba por si solo la funcionalidad que implementaba. Teniendo el pasante que acudir a los otros desarrolladores para la explicación de casos borde no apreciables en el análisis semántico. Además, comentarios y nombres de rutinas se encontraban escritos en alemán, idioma que el pasante no dominaba al nivel necesario para su completo entendimiento.
- El manejo de la internacionalización del sistema como fechas, husos horarios y lenguajes fue un reto interesante que el pasante debió sortear, siendo los clientes de diversos países dentro de la Unión Europea.
 - Para el manejo de fechas y husos horarios se aprovechó la interacción de los campos tipo *datetime* de SQL Server y la clase *DateTime* de PHP, que permitió configurar las locales dependiendo de las características del servidor donde se encuentre alojado el sistema.
 - Para el manejo del texto en diferentes lenguajes el pasante manejó una base de datos de la empresa donde se encuentran todos los textos de la aplicación. Insertando y consultando los textos adecuados en cada ocasión. Los textos de la base de datos luego pasaban a ser parte de una función filtro de todos los textos del sistema. Dicha función detectaba el idioma guardado en la sesión HTTP y mostraba el texto oportuno. El pasante ingresó todos los textos en inglés que luego fueron traducidos por colegas de la empresa.

CONCLUSIONES Y RECOMENDACIONES

La implementación e implantación de las funcionalidades de este proyecto de pasantías concluyó exitosamente y en el tiempo estipulado, el pasante se integró satisfactoriamente al equipo de programación de la empresa Fischer, Knoblauch & CO (FKC) con los que compartió durante el desarrollo. Derivando en que módulos para el manejo de seminarios; presenciales y en línea, así como un módulo para el soporte de las ubicaciones donde estos seminarios se realizan fueran agregados al Sistema de Gestión de Aprendizaje (SGA) base de FKC y al SGA de Bibliomed, uno de los clientes de la empresa. Permitiendo a los usuarios de dichos sistemas más flexibilidad a la hora de idear planes de estudios dentro del sistema, integrando soporte para el *b-learning*.

El proyecto fue realizado bajo la metodología SCRUM, la cual permitió que su desarrollo fuese planificado, eficiente y organizado, llevando a cabo los cambios necesarios para adaptarse a los nuevos requerimientos que constantemente fueron surgiendo a lo largo del mismo. El proyecto estuvo comprendido por diez sprints en donde se fueron desarrollando las historias de usuario que surgieron del levantamiento inicial de los requerimientos. Ordenadas en un marco de desarrollo de abajo hacia arriba donde el pasante intentó construir primero las funcionalidades más pequeñas, para luego construir con ellas las más grandes.

El desarrollo de los módulos se realizó bajo el patrón de arquitectura Modelo Vista Controlador (MVC), con la meta de mantener una estructura clara dentro del código que pueda ser mantenida y extendida por futuros desarrolladores. Durante la implementación del proyecto de pasantía, se utilizaron herramientas como el lenguaje de programación PHP: *Hypertext Preprocessor* (PHP), *Structured Query Language*. Lenguaje de consulta estructurado (SQL) y JavaScript para lograr los objetivos planteados.

Algunas recomendaciones surgen de la interacción del pasante con el SGA producido por FKC.

Se recomienda el uso de un manejador de versiones por parte del equipo de programación. Los beneficios podrían ser incalculables para la empresa pues este tipo de herramientas además de ser casi de uso obligado en el desarrollo de *software* desde hace más de quince años traen consigo una serie de bondades difíciles de rechazar, más aún si se toma en cuenta el *workflow* de FKC. Entre estas:

- Manejo de distintos ambientes de trabajo (desarrollo, producción, prueba) y sus

respectivas versiones que pueden ser compartidas por todos los desarrolladores de la empresa.

- Mejor gestión del equipo, ofrece una interfaz centralizada para entender quien hizo que, cuándo y cómo.
- Facilita la construcción de mejores estimaciones para el desarrollo de los proyectos.
- Una importante reducción del tiempo en la implantación del sistema comparándola con el servicio de archivos manualmente a través de *File Transfer Protocol*. Protocolo de transferencia de archivos (FTP).
- Soporte de la base de código externo a la empresa, haciéndola independiente del *hardware* interno.
- Eliminar la práctica de realización de *bakups* antes de cualquier cambio importante al sistema que consumen tiempo de los desarrolladores y espacio de disco en los servidores.
- Independencia de las bases de código de desarrollo y producción, donde desarrollo nuevo para el sistema no obstruya la implantación de un sistema nuevo o requiera el trabajo en una copia distinta de la base de código.

Si a todos estos beneficios le agregamos que FKC es una empresa que sirve a por lo menos quince otras, estos beneficios se multiplican, dado que existen distintas bases de código, repartidas en varios servidores a los que constantemente se realizan actualizaciones. Entre estas herramientas se encuentran GIT, Subversion, entre otras.

Se recomienda el uso de manejo de dependencias dentro del lenguaje de programación en los distintos proyectos de la empresa para el correcto manejo de los ambientes de desarrollo, entre estas herramientas se encuentran *composer* para PHP y NodeJS *Package Manager*. Manejador de paquetes de NodeJS (NPM).

En tiempos actuales, donde la globalización impera el pasante considera importante el desarrollo del sistema completamente en el idioma inglés. De esta forma se amplía fácilmente la cantidad de personas empleables por la empresa e incluso facilita la implementación de personal de apoyo fuera de las fronteras alemanas, dados los altos salarios que se necesitan para mantener un programador en dicho país.

Se recomienda una separación de los roles de diseñador y programador, haciendo uso eficiente del departamento de diseño gráfico dentro de la empresa. El pasante considera

una buena práctica generar primero un diseño de como lucirá la interfaz y como serán las interacciones, para que luego la funcionalidad sea plasmada por el programador. Mezclar estos dos trabajos crea fases de espera entre el desarrollo de lo que el programador cree es una interfaz válida y la revisión por parte del departamento gráfico, especialmente cuando no están claros los requerimientos inicialmente.

Refiriéndose a la base de código el pasante ofrece las siguientes recomendaciones:

El largo de muchos de los archivos y de muchas de las funciones dentro del sistema es excesivo. Esto complica a los programadores trabajar en conjunto en un mismo proyecto, así como la legibilidad del código y aumentando la cohesión del mismo. De esta forma es difícil que un pequeño cambio no afecte alguna otra funcionalidad. Es además una de las primeras pistas que evidencian la mezcla de la lógica del negocio con el rol de los modelos y la interfaz definidos en el patrón MVC. Todo esto hace que la curva de aprendizaje de un nuevo desarrollador sea aún más empinada.

El código en su mayoría no es reusable, pedazos de código que realizan la misma función se ven en distintos archivos, replicando conocimiento y haciendo imposible la mantenibilidad eficiente del sistema. Se recomienda el uso del patrón MVC o en su defecto de cualquier otro patrón que se considere oportuno luego de un análisis de las funcionalidades del sistema. Además, se recomienda el uso de patrones de diseño, que favorezca la estructuración del código, promoviendo así la reusabilidad y facilitando el trabajo en equipo y las revisiones por parte del jefe de proyectos.

La lógica de negocio se encuentra mezclada con el acceso a los modelos y a la interfaz, haciendo imposible la asignación de desarrolladores a distintas tareas y eliminando la oportunidad de integrar a un experto en interfaces al equipo de desarrollo del sistema, es decir, cambiar la interfaz es casi imposible sin cambiar como se acceden a los datos o interferir en las reglas del negocio. Recordando que la personalización visual del sistema para cada cliente es una de las características más importantes para FKC.

La implementación de pruebas a lo largo del sistema. El sistema no contiene pruebas que lo sustenten. No tenerlas trae consigo un conjunto de problemas en el desarrollo de *software*. Al crear una nueva funcionalidad es conveniente ejecutar las pruebas para asegurarse que ninguna otra funcionalidad es afectada, esto combinado con los puntos antes descritos es caldo de cultivo para la proliferación de errores dentro del sistema. La realización de pruebas puede consumir tiempo al inicio del desarrollo, pero a medida que el sistema crece la retribución de ese tiempo invertido crece exponencialmente, pues elimina

el tiempo de prueba que se consume luego de cada cambio importante, por ejemplo, un cambio de versión de PHP, entre otros.

Eliminar el uso de *iframes Hypertext Markup Language*. Lenguaje de marcado de hipertexto (HTML) para el desarrollo de la interfaz, esta es una técnica ampliamente criticada pues dificulta el manejo de la navegación dentro del sistema y no aprovecha el uso de elementos ya cargados en las páginas donde estos se llaman. Limitando su uso a la demostración de contenido multimedia u otras páginas fuera del sistema, para lo que estos fueron construidos.

Promover el uso de peticiones del tipo *Asynchronous JavaScript And XML*. JavaScript y XML asíncronos (AJAX) (menos recargas completas de las páginas dentro del sistema), implementación de *Application Programming Interface*. Interfaz de programación de la aplicación (API) internas e implementar manejo de *cache* para las consultas a la base de datos, reduciendo así el trabajo de los servidores web y de bases de datos o el futuro costo de *hosting* si se decide implementar soluciones en la nube. Aumentando la cantidad de clientes atendidos con las capacidades actuales de la empresa.

Por último, el sistema no posee documentación alguna, exceptuando las autogeneradas por algunas de las herramientas y no se invierte tiempo en mantenerlas. Esto combinado con la estructura del código complica el desarrollo de *software*.

El pasante por lo tanto recomienda una reestructuración y reescritura completa del SGA base de FKC. Que bien podría seguir siendo en PHP o podría explorar algunas tecnologías nuevas. Que pueda aprovechar las bondades de un *framework* para PHP como Laravel o Symphony. Estos frameworks facilitan la construcción de una estructura que permite el reúso de código, el trabajo en equipo y recortan los tiempos de desarrollo. Estas cualidades son muy importantes para FKC, ya que constantemente se encuentran desarrollando individualmente para algunos de sus clientes y poder portar las funcionalidades realizadas de un cliente a otro puede liberar tiempo para la creación de nuevas funcionalidades.

Para esta reestructuración se recomienda el uso de una metodología de desarrollo de *software*, que permita un desarrollo planificado, eficiente y organizado. que no descuide la realización de pruebas de *software* automatizadas y tenga como pilar fundamental el trabajo colaborativo basado en una herramienta de control de versiones correctamente usada.

REFERENCIAS

- [1] Sebastian Bergmann. *Página oficial de PHP Unit*. 2017. URL: <https://phpunit.de/> (visitado 25-08-2017).
- [2] Blackboard. *Página oficial de Blackboard*. 2017. URL: <http://www.blackboard.com/> (visitado 21-09-2017).
- [3] Equipo de bootstrap. *Página oficial de bootstrap*. 2017. URL: <http://getbootstrap.com/> (visitado 25-08-2017).
- [4] Steve Faulkner y col. *HTML 5.2*. 2017. URL: <https://w3c.github.io/html/> (visitado 25-08-2017).
- [5] Norm Friesen. *Report: Defining Blended Learning*. 2002. URL: http://learningspaces.org/papers/Defining_Blended_Learning_NF.pdf (visitado 22-08-2017).
- [6] The Apache Software Foundation. *Apache HTTP Server Project*. 2017. URL: <https://httpd.apache.org/> (visitado 25-08-2017).
- [7] Fundación jQuery. *Documentación jQuery*. 2017. URL: <https://blog.jquery.com/> (visitado 25-08-2017).
- [8] Moodle. *Página oficial de Moodle*. 2017. URL: <https://moodle.org/> (visitado 21-09-2017).
- [9] Equipo de la fundación Mozilla. *AJAX. Primeros Pasos*. 2017. URL: https://developer.mozilla.org/es/docs/AJAX/Primeros_Pasos (visitado 25-08-2017).
- [10] Equipo de la fundación Mozilla. *CSS*. 2017. URL: <https://developer.mozilla.org/es/docs/Web/CSS> (visitado 25-08-2017).
- [11] Equipo de la fundación Mozilla. *JavaScript*. 2017. URL: <https://developer.mozilla.org/es/docs/Web/JavaScript> (visitado 25-08-2017).
- [12] Inc. New Digital Group. *Página oficial de Smarty*. 2017. URL: <https://www.smarty.net/> (visitado 25-08-2017).
- [13] El grupo PHP. *Página oficial de PHP*. 2017. URL: <https://secure.php.net/> (visitado 25-08-2017).
- [14] Dirk Riehle. «Framework Design A Role Modeling Approach». Tesis doct. SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH, 2000.

- [15] Scrum.Org y ScrumInc. *The Scrum Guide*. 2017. URL: <http://www.scrumguides.org/scrum-guide.html> (visitado 29-08-2017).
- [16] s.f. *Arquitectura Cliente Servidor*. URL: https://www.ecured.cu/Arquitectura_Cliente_Servidor (visitado 22-08-2017).
- [17] Equipo de Swift Mailer. *Página oficial de Swift Mailer*. 2017. URL: <https://swiftmailer.symfony.com/> (visitado 25-08-2017).
- [18] Micheal Szabo y K Flesher. *CMI Theory and Practice: Historical Roots of Learning Management Systems*. Montreal, Canada: Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, e Higher Education, 2002.
- [19] Luis A Pérula de Torres. *El aprendizaje electrónico (e-learning, b-learning)*. 2011. URL: <http://www.doctutor.es/2011/12/02/ideas-y-reflexiones-en-educacion-medica-diciembre-2011/> (visitado 22-08-2017).
- [20] R. Trygve y J. O. Coplien. *The DCI Architecture: A New Vision of Object-Oriented Programming*. 2009. URL: http://www.artima.com/articles/dci_vision.html (visitado 22-08-2017).
- [21] Wikipedia. *Database*. 2017. URL: <https://en.wikipedia.org/wiki/Database> (visitado 25-08-2017).
- [22] Wikipedia. *Microsoft SQL Server*. 2017. URL: https://en.wikipedia.org/wiki/Microsoft_SQL_Server (visitado 25-08-2017).

APÉNDICE A

DIAGRAMAS

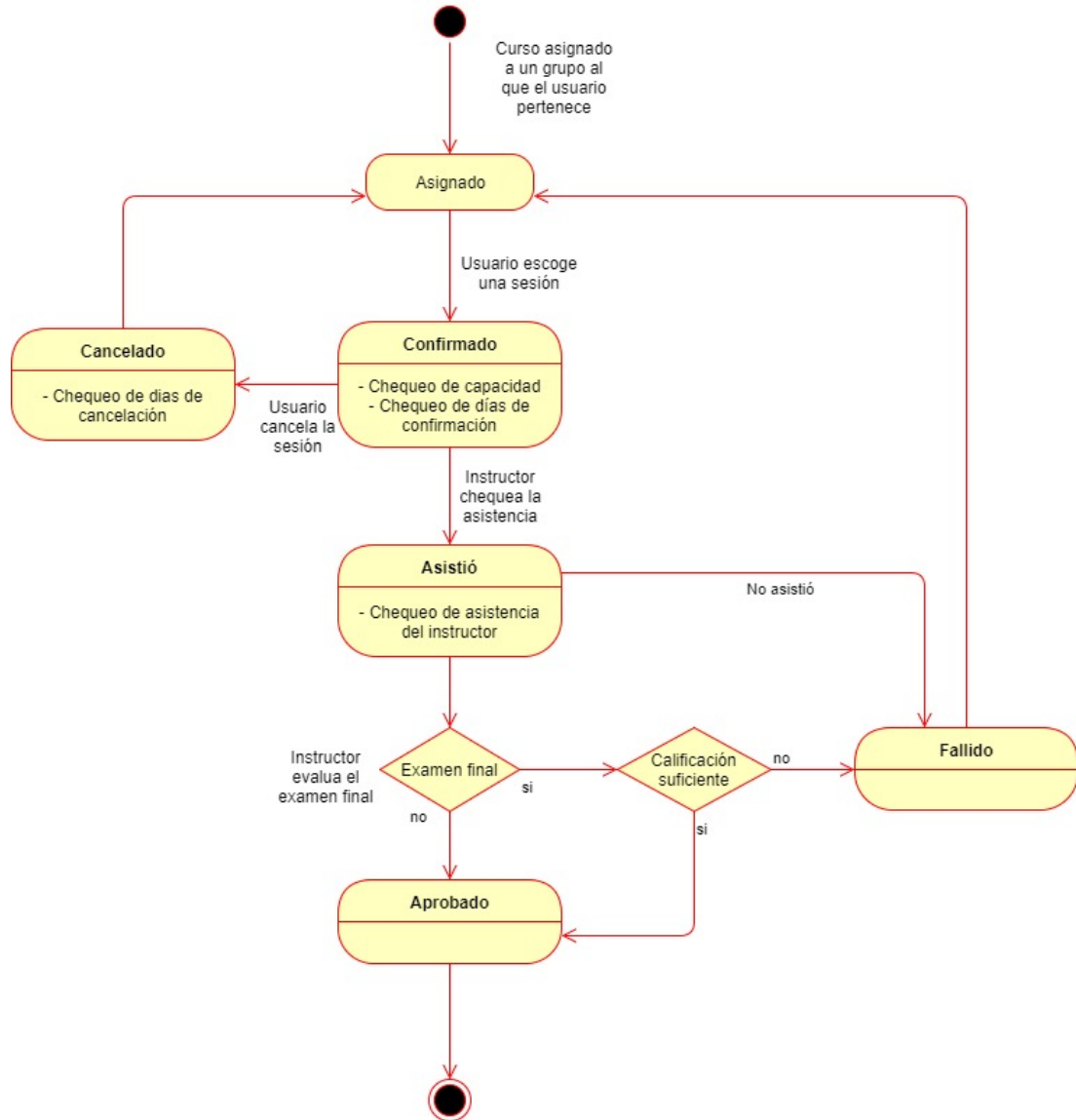


Figura A.1: Diagrama que demuestra los distintos estados en los que se puede encontrar una sesión presencial.

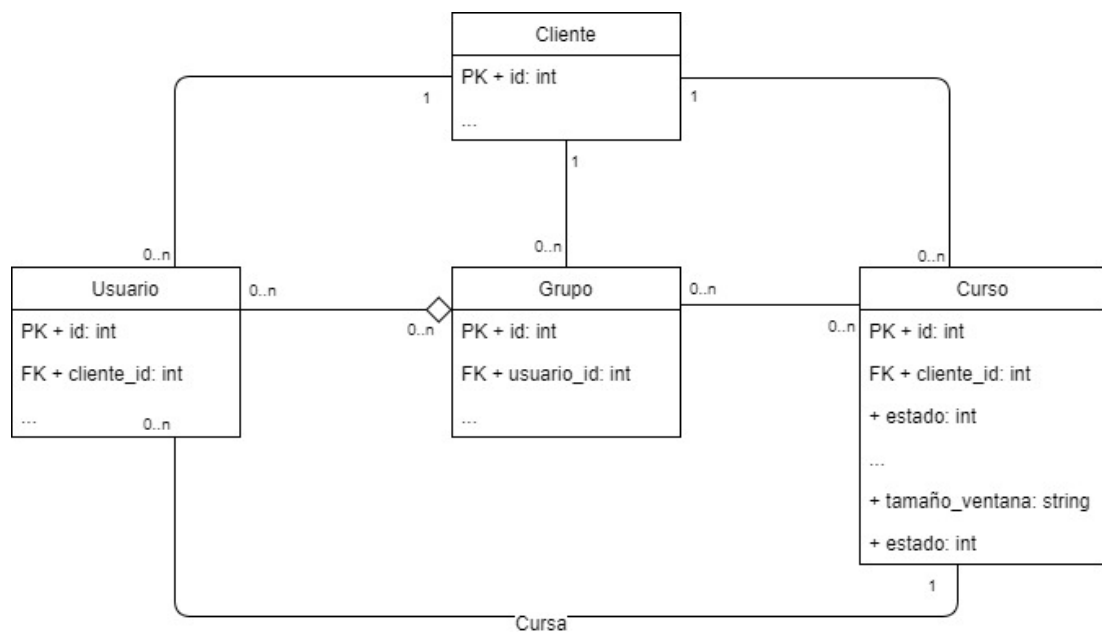


Figura A.2: Diagrama UML parcial de la base de datos del SGA de FKC previa al proyecto.

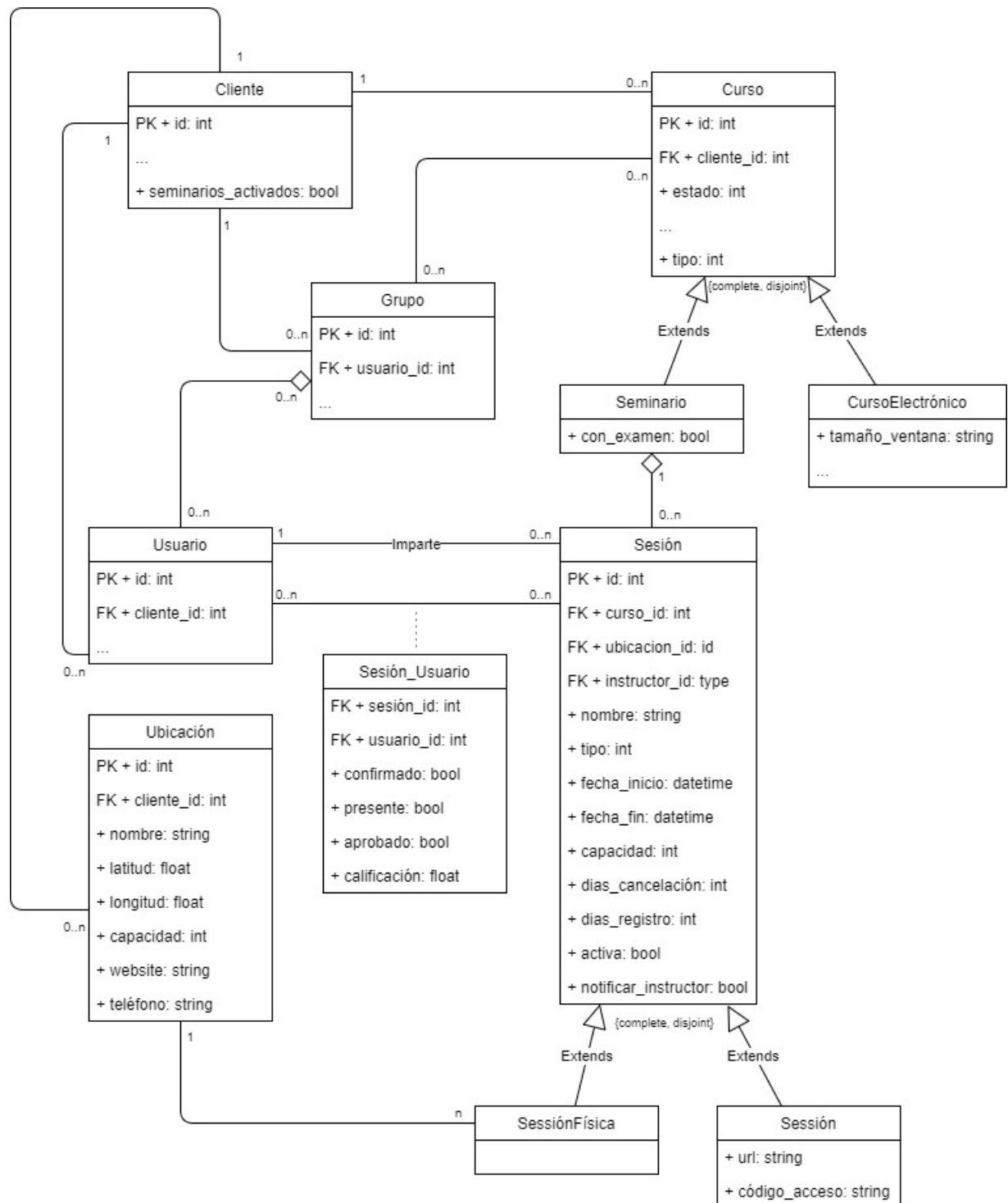


Figura A.3: Diagrama UML parcial de la base de datos final del SGA de FKC.

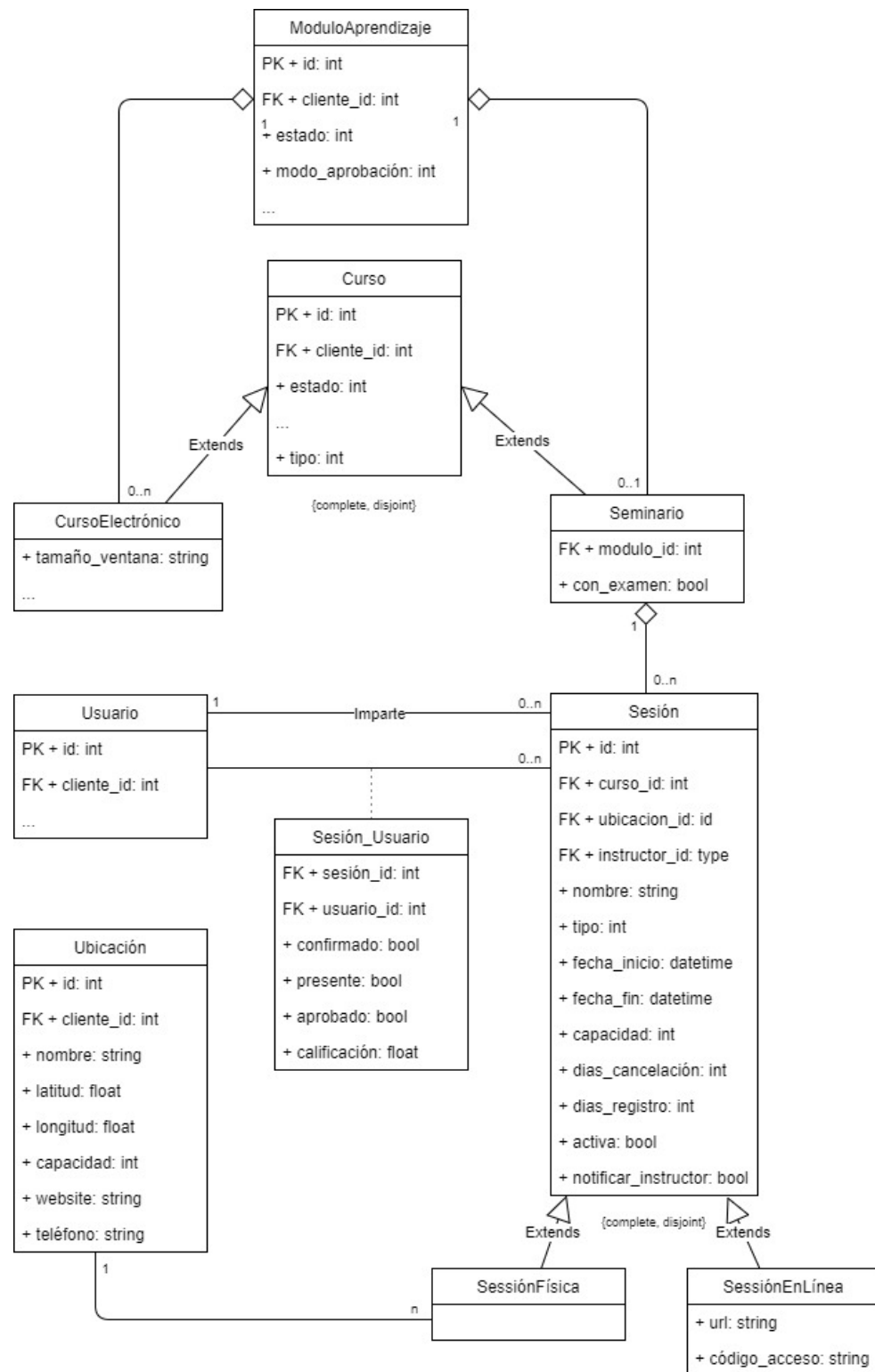
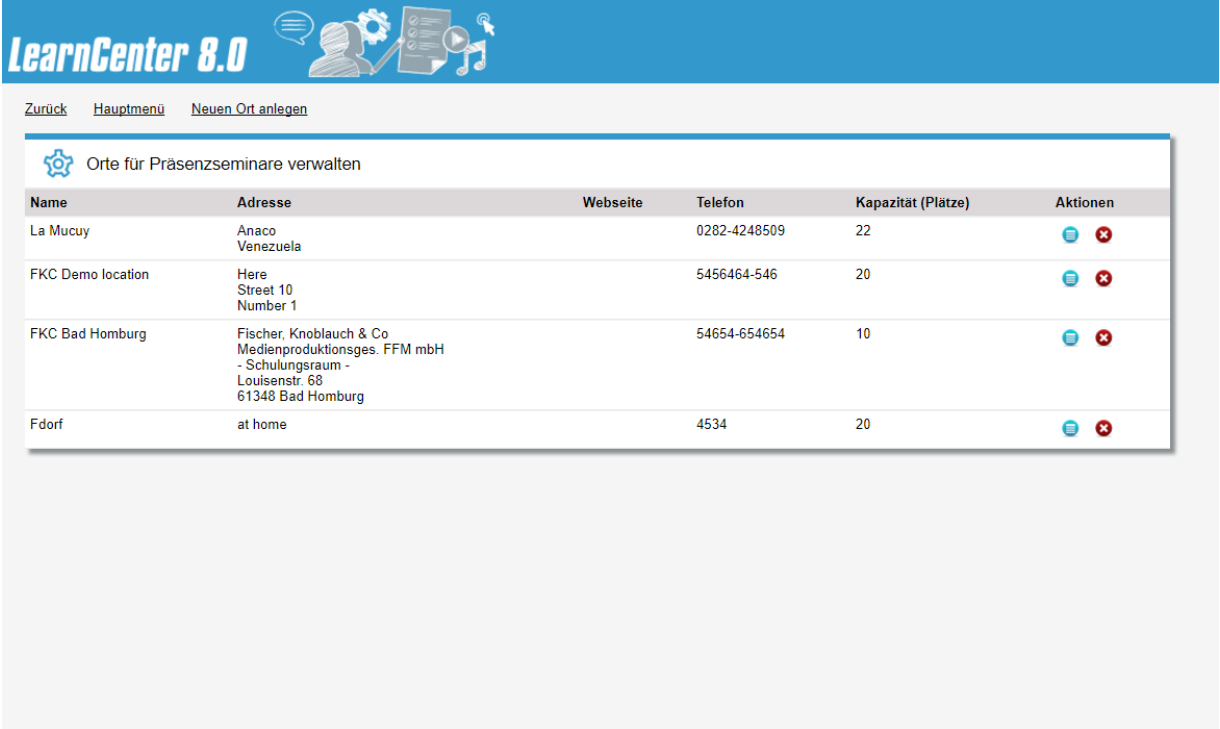


Figura A.4: Diagrama UML parcial de la base de datos final del SGA de Bibliomed.

APÉNDICE B

SCREENSHOTS DE LOS SISTEMAS



The screenshot displays the 'LearnCenter 8.0' web application. At the top, there is a blue header with the logo and navigation links: 'Zurück', 'Hauptmenü', and 'Neuen Ort anlegen'. Below the header, a section titled 'Orte für Präsenzseminare verwalten' (Manage locations for presence seminars) contains a table with the following data:









Name	Adresse	Webseite	Telefon	Kapazität (Plätze)	Aktionen
La Mucuy	Anaco Venezuela		0282-4248509	22	 
FKC Demo location	Here Street 10 Number 1		5456464-546	20	 
FKC Bad Homburg	Fischer, Knoblauch & Co Medienproduktionsges. FFM mbH - Schulungsraum - Louisenstr. 68 61348 Bad Homburg		54654-654654	10	 
Fdorf	at home		4534	20	 

Figura B.1: Vista del listado de Ubicaciones.

LearnCenter 8.0

Zurück Hauptmenü

Ort bearbeiten

Name: La Mucuy

Adresse: Anaco, Venezuela

Webseite:

Telefon: 0282-4245500

Kapazität (Plätze): 22

Google Maps URL:

[Informationen](#)

9°26'13.5"N 64°29'02.5"W

[View larger map](#) [Directions](#) [Save](#) [Sign in](#)

Präsenzseminartermine geplant für La Mucuy

2015 2016 2017 2018 2019

Januar

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Februar

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5			
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28					

Marz

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5			
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

April

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Mai

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5	6	7	
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Juni

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4				
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

Juli

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

August

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

September

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Oktober

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

November

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Dezember

Mo	Di	Mi	Do	Fr	Sa	So
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					


Jose Pascarella 14:45

Figura B.2: Vista de la edición de una ubicación.

LearnCenter 8.0



[Zurück](#) [Hauptmenü](#)

 **Kurs anlegen**

Typ

Präsenzseminar ▼

Kursname

Sprache

☒ de
☐ en

Vorschaubild

Choose File

No file chosen

Hinweis:
Unterstützte Bildformate sind .png, .jpg und .gif.
Die empfohlene Bildgröße ist 200 x 200 Pixel.

Themengebiet

Allgemeine Themen ▼

Art der Teilnahmebestätigung

☒ Anwesenheit
☐ Anwesenheit und Test

Kursbeschreibung

Übernehmen

Figura B.3: Vista de la creación de un seminario.

LearnCenter 8.0

Hauptmenü Neuen Kurs anlegen Kurse suchen Kurs(e) kopieren SCORM-Kurs importieren

Vorhandene Kurse

Themengebiet: Allgemeine Themen nur Präsenzseminare ☒

Kurs_ID	Kurs	Themengebiet / Kursbeschreibung / Rechtevergabe	Typ
<input type="checkbox"/> 172	Hygieneschulung 2	Allgemeine Themen TEST	Präsenzseminar
<input type="checkbox"/> 158	New Rewe Seminar	Allgemeine Themen Used for testing de seminar module	Präsenzseminar
<input type="checkbox"/> 163	Online Meetings Seminar	Allgemeine Themen This is for testing	Präsenzseminar

Figura B.4: Vista del listado de cursos del sistema para el administrador.

Fenster schließen

Kurse der Gruppe: Ebene 2


Themengebiet: Allgemeine Themen nur Präsenzseminare ☒

Kurs_ID	Kurs (Beschreibung)	Vorgabe-Anfang	Vorgabe-Ende
<input type="checkbox"/> 172	Hygieneschulung 2 (TEST)	Zu diesem Präsenzseminar wurden keine Termine angelegt.	
<input checked="" type="checkbox"/> 158	New Rewe Seminar (Used for testing de seminar module)	Zu diesem Präsenzseminar wurden keine Termine angelegt.	
<input type="checkbox"/> 163	Online Meetings Seminar (This is for testing)	Präsenzseminartermin	Startdatum
		Sesion de prueba	10.08.2017, 14:45


Übernehmen

Figura B.5: Vista de la asignación de un seminario a un grupo.

LearnCenter 8.0



[Zurück](#) [Hauptmenü](#)

 **Präsenzseminartermine verwalten**





Name	<input type="text" value="Jose Pascarella"/>	
Typ	Vor-Ort	
Ort 	<input type="text" value="La Mucuy"/> ▼	
Trainer	<input type="text" value="Kümmel (FKC ffm), Peer"/> ▼	
Startdatum	<input type="text" value="10.08.2017 14:45"/>	
Endedatum	<input type="text" value="11.08.2017 15:00"/>	
Kapazität (Plätze)	<input type="text" value="22"/>	
Registrierung möglich bis x Tage vor Startdatum	<input type="text" value="0"/>	
Stornierung möglich bis x Tage vor Startdatum	<input type="text" value="0"/>	
<input type="checkbox"/> Benachrichtigungs-E-Mails an den Trainer senden		
<input checked="" type="checkbox"/> Aktiv		
<input type="button" value="Übernehmen"/>		

Figura B.6: Vista de la creación de una sesión de un seminario.

LearnCenter 8.0

[Zurück](#) [Hauptmenü](#) [Neuen Präsenzseminartermin anlegen](#)

Präsenzseminartermine für: Online Meetings Seminar

Präsenzseminartermin	Ort / Online-Meeting	Trainer	Startdatum	Endedatum	Anmeldungen	Aktiv	Aktionen
tärmün	La Mucuy	Christian Ament	23.07.2017, 15:15	23.07.2017, 15:30	0	<input type="checkbox"/>	+ - x

Figura B.7: Vista del listado de las sesiones de un seminario.

LearnCenter 8.0

[Zurück](#) [Hauptmenü](#)

Teilnehmerliste: Morning session

Name	E-Mail-Adresse	Telefon	Besucht Alle besucht	Bestanden Alle bestanden
Pascarella, Jose	josepas979@gmail.com		<input checked="" type="checkbox"/>	<input type="checkbox"/>




Figura B.8: Vista de los cursos de un instructor.

Teilnehmerliste			
Präsenzseminar:	New Rewe Seminar	Präsenzseminartermin:	Sesión matutía
Datum:	20.08.2017, 15:45 / 20.08.2017, 17:00	Trainer:	Jose Pascarella
Name		Besucht	
Pascarella, Jose		<input type="checkbox"/>	

Figura B.9: Archivo PDF generado con la lista de los estudiantes de un curso.

LearnCenter 8.0

Ebene 1 Ebene 2 Eins Präsenzseminare Rossmann

Kurse

Themengebiet Kurse Status Start

Allgemeine Themen	► New Rewe Seminar	<input type="checkbox"/>	
	► Online Meetings Seminar	<input type="checkbox"/>	
SAG	► New SAG Seminar	<input checked="" type="checkbox"/>	X

Impressum
Administration
English

Figura B.10: Vista de los cursos disponibles para el usuario aprendiz.

LearnCenter 8.0

Ebene 1 Ebene 2 Eins Präsenzseminare Rossmann schwarz/Lidl/Kaufland Seminar testing group Webentwickler Alle Teilnehmer

Kurse

Themengebiet Kurse Status Start

Rossmann	► Durchführung der Kassiertätigkeit Teil 1	<input type="checkbox"/>	X
	► Durchführung der Kassiertätigkeit Teil 2	<input type="checkbox"/>	X
	► Vorbereitung der Kassiertätigkeit	<input type="checkbox"/>	X
	► Vorbereitung der Kassiertätigkeit (Videotest 21.09.2016)	<input type="checkbox"/>	X

Umfragen
Erste Umfragevorlage

Impressum
Administration
English

Figura B.11: Vista de los cursos disponibles para el usuario aprendiz antes de la extensión realizada en la pasantía.

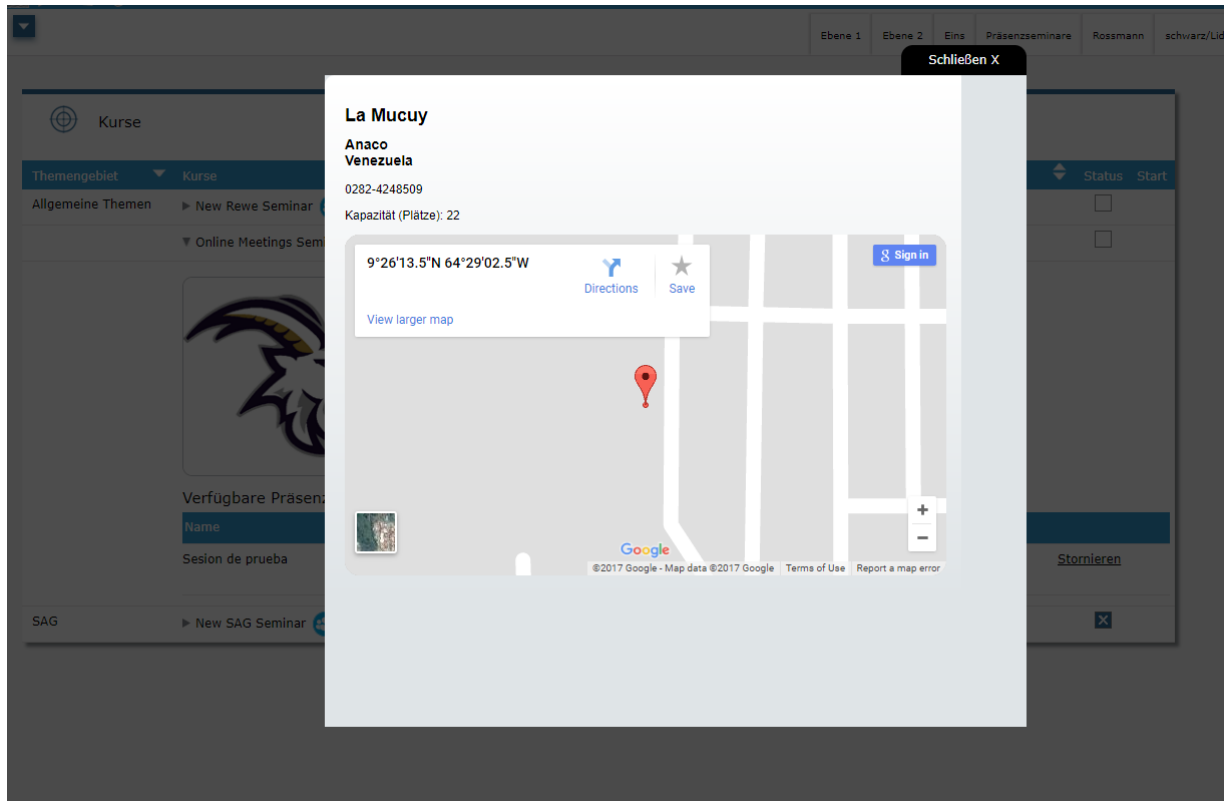


Figura B.12: Vista del modal mostrado con los datos de la ubicación de la sesión correspondiente.

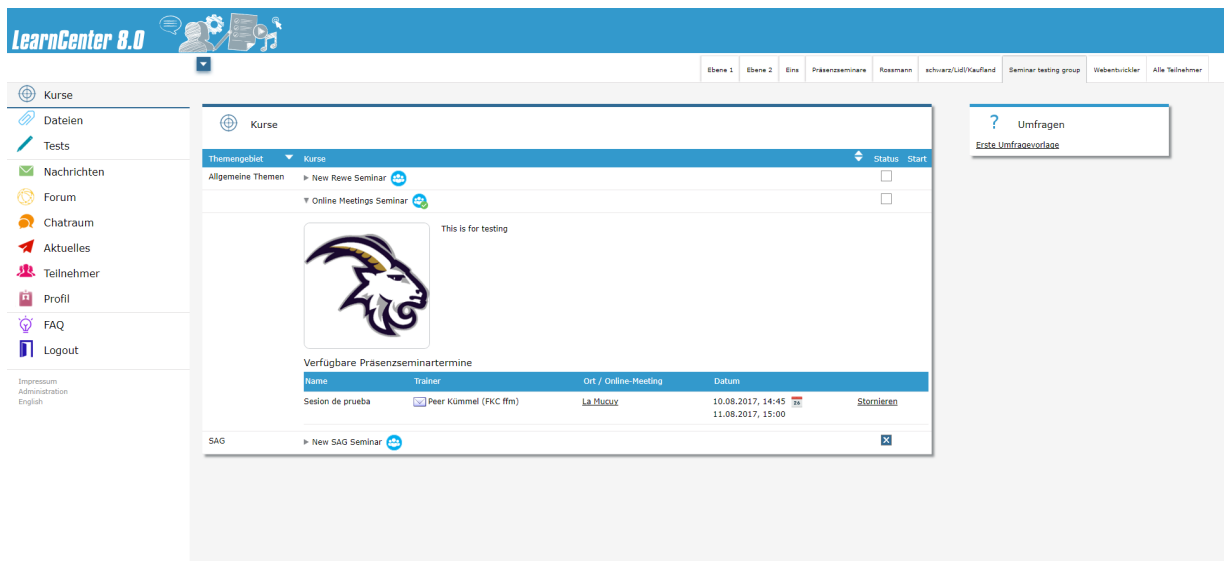


Figura B.13: Vista de una sesión confirmada por el usuario.

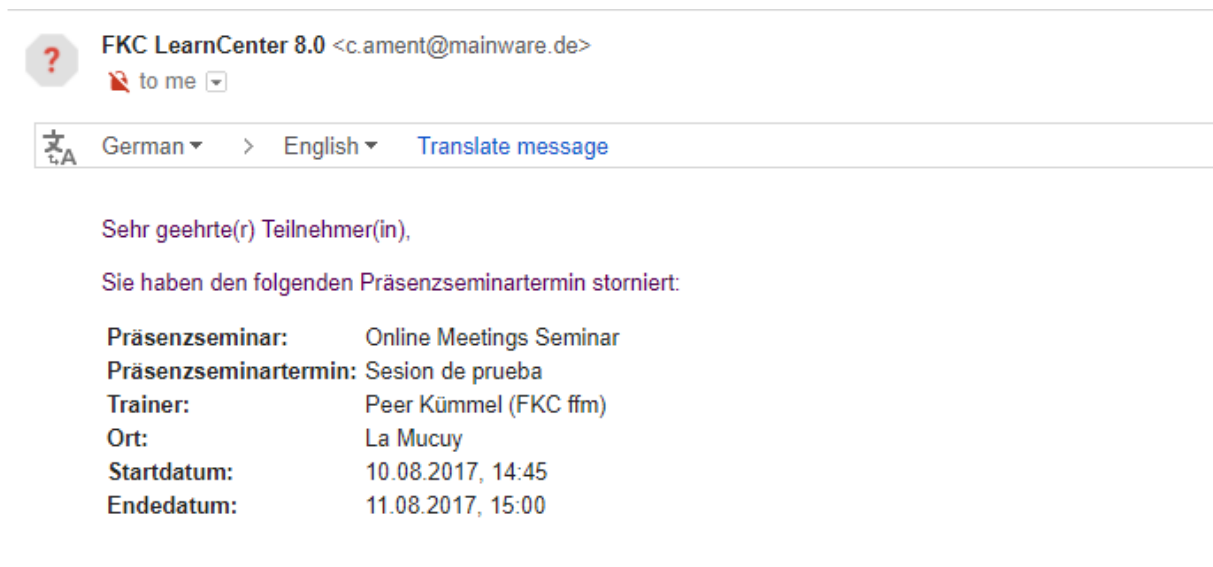


Figura B.14: Formato de los correos enviados por el sistema.

LearnCenter 8.0

Hauptmenü Gesamt-Statistik Kurs-Statistik Test-Statistik Login-Statistik

Kurs-Statistik

Kursübersicht nach Gruppe/Kurs/Teilnehmer


Gruppe: [Filter zurücksetzen](#)


Kurs:


Teilnehmer: Ergebnisse:

Name	Personalnummer	Kurs	Status	abgeschlossen	Erster Kursstart	Letzter Kursstart	Gesamtdauer
Ament, Christian		Basisqualifizierung Partner: 1 Willkommen	<input type="checkbox"/>		21.03.2016, 13:55	21.03.2016, 14:02	00:06:43
		Dieben keine Chance	<input type="checkbox"/>		21.04.2015, 15:57	04.11.2015, 15:10	00:01:16
		F - Fashion Master	<input type="checkbox"/>		03.07.2015, 10:54	04.11.2015, 14:46	00:01:40
		F - Fashion Master RU	<input type="checkbox"/>		03.07.2015, 11:13	03.07.2015, 11:13	00:00:20
		New Rewe Seminar	<input type="checkbox"/>		--	--	--
Benutzer, Einfacher		Basisqualifizierung Partner: 1 Willkommen	<input type="checkbox"/>		--	--	00:00:00
		Dieben keine Chance	<input type="checkbox"/>		--	--	00:00:00
		F - Fashion Master	<input type="checkbox"/>		--	--	00:00:00
		F - Fashion Master RU	<input type="checkbox"/>		--	--	00:00:00
		New Rewe Seminar	<input type="checkbox"/>		--	--	--

Figura B.15: Vista de las estadísticas mostradas al administrador.

[Back](#) [Main menu](#)

 Edit module

Learning module	<input type="text" value="Module with Seminars"/>
Topic	<input type="text" value="Demo"/>
ID	28
Active	<input checked="" type="checkbox"/>
Module approval	<input checked="" type="radio"/> E-learning or Seminar <input type="radio"/> E-learning and Seminar
RbP course ID	<input type="text"/>
Availability	<input type="checkbox"/> Cross-customer 

Save

General

Learning units

Files

Seminar


Manage sessions

☒ Seminars enabled

Name	<input type="text" value="This is another"/>
Language	en
Way of approval	<input checked="" type="radio"/> Attendance
Description	<div><div>This is the seminar descriptions</div></div>

Save

Figura B.16: Vista de la adaptación hecha para la creación de seminarios en el SGA de Bibliomed.



Lernmodule

Tests

Zertifikate

Dateien

Links

Nachrichten

Forum

Chatraum

Aktuelles

Termine

FAQ

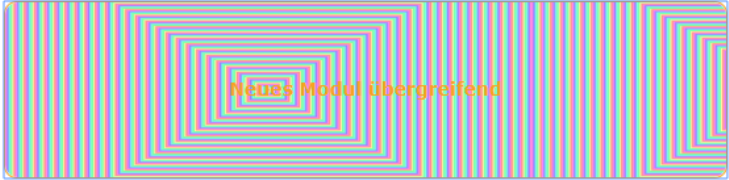
Profil

Logout


Impressum
Administration
English
Русский язык


Lernmodule (Vorschau)


- freigegebenes Themengebiet



Inhalt dieses Moduls
Modul zum testen, übergreifend freigegeben.


E-Learning


Zur Vertiefung



Zertifikat

Nordamerika

+ Thema für GNH

+ Seminars

Figura B.17: Vista de cursos disponibles para el aprendiz antes de la integración de el módulo de seminarios en el SGA de Bibliomed.



Lernmodule

Tests

Zertifikate

Dateien

Links

Nachrichten

Forum

Chatraum

Aktuelles

Termine

FAQ

Profil

Logout

Impressum

Administration

English

Русский язык

Lernmodule (Vorschau)


+ freigegebenes Themengebiet


+ Thema für GNH


- Seminars


Module with Seminars

Inhalt dieses Moduls











Verfügbare Präsenzseminartermine

Name	Trainer	Ort / Online-Meeting	Datum
First online session	 Christian Ament	Online-Meeting	04.08.2017, 17:15 05.09.2017, 17:30
Another Session	 Einfacher Benutzer	FKC Demo location	05.10.2017 12:30 - 14:00


[Stornieren](#)  26

Figura B.18: Vista de una sesión reservada en el SGA de Bibliomed.