



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

SISTEMA DE APOYO A LA CONSTRUCCIÓN Y MANTENCIÓN DE MÓDULOS
EDUCATIVOS

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN COMPUTACIÓN

FRANCO ALBERTO CRUCES AYALA

PROFESOR GUÍA:
SERGIO OCHOA DELORENZI

MIEMBROS DE LA COMISIÓN:
JOSÉ PINO URTUBIA
EDUARDO GODOY VEGA

SANTIAGO DE CHILE
2020

Resumen

LeanSight es una consultora de innovación y tecnología que ofrece formaciones y talleres orientados a mejora de la cultura organizacional. Los talleres se implementan principalmente en archivos de presentación de diapositivas y documentos de texto, interconectados de forma compleja y poco legible durante su diseño. Los recursos se modifican varias veces por semana con cambios que podrían tomar horas en implementarse con el objetivo de la mejora continua.

Este trabajo explora la problemática sobre la mantención y creación de contenido educativo mediante la inmersión en los quehaceres cotidianos de la organización, y a partir de un análisis riguroso de la situación actual, propone un modelo lógico para la estructura de los módulos educativos, implementando una plataforma que habilita el trabajo en torno al modelo planteado. Finalmente se evalúa la efectividad de la solución con un caso de uso real.

El modelo propuesto corresponde a una forma de interpretar las modificaciones realizadas sobre los módulos educacionales, distinguiendo tres capas lógicas en las que pueden ocurrir: (1) el planteamiento de un cambio de paradigma sobre los participantes de los talleres, (2) habilidades específicas alineadas con el cambio de paradigma propuesto y (3) cada actividad educativa con intenciones concretas basadas en pedagogía para entregar, en conjunto, las habilidades perseguidas. Se entiende que los archivos concretos que se manejan son simplemente visualizaciones del concepto de actividad y no es cada uno su propia entidad. A partir de esto y las observaciones anteriores, se plantea el desafío focal de crear una conexión persistente y mantenible entre las actividades, sus intenciones y las visualizaciones de cada una.

La plataforma implementada quita el foco del trabajo en los archivos de un módulo y lo lleva a las actividades educativas que lo conforman, apoyándose en el modelo propuesto para dar sentido a las modificaciones y automatizar varias tareas que normalmente serían muy costosas. Los archivos finales se generan a partir de Markdown, implementaciones legibles, mantenibles y automatizables, para habilitar prototipos rápidos y evitar la pérdida de foco.

Se pone a prueba la solución con un módulo educativo real, implementado por un miembro del equipo sin experiencia previa con la aplicación. Se verifica que la solución entrega una mejora considerable a las problemáticas identificadas, reduciendo en gran medida la cantidad de tiempo necesaria para realizar modificaciones que se implementan con frecuencia. El usuario se siente cómodo con la aplicación y reconoce las posibilidades que entrega, además de proponer cambios que siguen mejorando la usabilidad, utilidad y eficacia del software.

Finalmente, apoyándose en la investigación realizada, se identifican mejoras para la aplicación y se plantean como un conjunto de funcionalidades futuras por diseñar e implementar.

*A todos mis amigos y compañeros que lo pasan mal durante sus estudios...
no dejemos de apoyarnos y cuidarnos entre todos.*

Agradecimientos

Este documento representa el final de un ciclo, el final de una larga carrera, llena de dificultades y experiencias enriquecedoras. Pero todas estas experiencias no serían nada sin todas las personas que me acompañaron, me apoyaron y me ayudaron a sacar valiosos frutos de él.

Agradezco más que a nadie a mi familia, que siempre hizo todo lo que estuviera a su alcance para asegurar mi educación y mi bienestar.

Mi cálido agradecimiento a Karina, quien me ha acompañado en mis momentos más difíciles y me ha apoyado incondicionalmente cuando más lo he necesitado.

A mis amigos más cercanos, en quienes encontré un inmenso cariño y preocupación correspondidos, gracias por hacer este proceso mucho más agradable y entretenido. Aunque varios de ellos no estuvieron conmigo durante todo el camino, atesoro su cada momento con todos y aseguro que fueron parte esencial de esta aventura.

A mis compañeros de videojuegos, que nunca fallaron en divertirme y distraerme de situaciones abrumadoras, gracias por siempre ganar en diversión.

Gracias a todos mis compañeros de carrera, que saben todo lo que significa ser un estudiante y con quienes siempre compartí apoyo y cuidado.

A todos aquellos profesores profundamente comprometidos con su papel en nuestro crecimiento personal, gracias por su esfuerzo y por ser una inspiración para mí y muchos más.

Gracias a LeanSight y todos sus miembros por hacerme posible trabajar en algo que me entretiene y apasiona, además de motivarme a explorar ámbitos que desconocía.

Muchas gracias a todos quienes me apoyaron de una forma u otra durante este viaje lleno de risas y llantos, de caídas y aprendizajes. Gracias.

Tabla de Contenido

1. Introducción	1
1.1. Contexto	1
1.2. Problemática a abordar	2
1.3. Situación actual	4
1.3.1. Cuerpo de contenidos educativos	4
1.3.2. Soporte tecnológico	5
1.4. Objetivos de la memoria	6
1.5. Estructura del documento de memoria	7
2. Estudio de la problemática	8
2.1. Viviendo el proceso	8
2.1.1. Preparación de talleres	8
2.1.2. Realización de talleres	9
2.1.3. Modificación de talleres	10
2.1.4. Confección de talleres y módulos	11
2.2. Haciendo explícito el conocimiento del equipo	12
2.3. Buenas prácticas pedagógicas	13
2.4. Control de versiones	14
2.5. Revisión de sesiones y módulos	14
3. Flujos de trabajo	16
3.1. Actores y entidades	16
3.2. Modificación de recursos	18
3.3. Creación de sesiones	19
3.4. Flujo creativo	20
3.5. Síntomas y dolores identificados	20
4. Análisis de la situación actual	24
4.1. ¿Por qué se modifica?	24
4.2. ¿Qué se modifica?	25
4.3. Recursos de documentación	27
4.4. Reformulación del escenario de trabajo	28
4.5. Desafío focal	29
5. Concepción de la solución	30
5.1. Plataforma de edición	30
5.2. Generación de documentos	31

5.3. Estructura conceptual de la solución	33
5.4. Framework para la aplicación	34
5.5. Modelo de datos	35
6. Implementación de la solución	38
6.1. Navegación en la aplicación	38
6.2. Funcionalidades	38
6.2.1. Editor	38
6.2.2. Generador de documentos	39
6.2.3. Lista de actividades	40
6.2.4. Previsualización	42
6.2.5. Biblioteca de módulos	43
6.2.6. Guardado automático	44
6.2.7. Notificaciones	44
6.2.8. Administrador de imágenes	45
6.2.9. Insertar imágenes	45
6.2.10. Reordenar actividades	47
6.2.11. Seguridad	47
6.2.12. Ocultar actividades	47
6.3. Interfaz de usuario	48
7. Validación de la solución	50
7.1. Ciclos de retroalimentación	50
7.2. Prototipo de baja tecnología	51
7.3. Prueba realizada	52
7.4. Medición de impacto	53
7.5. Utilidad percibida	54
8. Conclusiones y trabajo a futuro	57
8.1. Lecciones aprendidas	58
8.2. Trabajo a futuro	58
Glosario	60
Bibliografía	61

Índice de Tablas

3.1. Estimaciones de tiempo al ejecutar modificaciones típicas	19
5.1. Evaluación de alternativas de editor de documentos	33
7.1. Tiempo estimado para ejecutar modificaciones típicas anterior a la implementación (Tiempo anterior) y posterior a la misma (Tiempo nuevo)	53

Índice de Ilustraciones

1.1. Ejemplo de diseño de módulo educativo	3
1.2. Algunos módulos educativos almacenados en Google Drive	5
3.1. Diagrama de actores y entidades relevantes	17
3.2. Flujo de modificación de módulos	22
3.3. Flujo de creación de sesiones	23
4.1. Modelo de tres capas para módulos educativos	26
4.2. Distinción entre la intención y la visualización de una actividad	26
5.1. Diagrama de interacciones entre actores y componentes de la aplicación	34
5.2. Modelo de datos de la solución	37
6.1. Mapa de navegación de la aplicación	39
6.2. Editor basado en Markdown	40
6.3. Ejemplo de diapositiva generada desde Markdown	41
6.4. Ejemplo de cuadernillo generado desde Markdown	41
6.5. Lista de intenciones para navegar el contenido	42
6.6. Ejemplo de previsualización en cuadernillo y diapositivas	43
6.7. Biblioteca de módulos	44
6.8. Notificación de éxito durante el guardado de cambios	45
6.9. Notificación de fracaso durante el guardado de cambios	45
6.10. Biblioteca de imágenes	46
6.11. Ejemplo de inserción de una imagen bajo el identificador <i>filas</i>	46
6.12. Ejemplo de actividades ocultas, la primera y segunda (seleccionadas) se encuentran ocultas y la tercera no	48
6.13. Interfaz de edición completa	48
7.1. Primeras intenciones de uno de los prototipos realizados	51
7.2. Flujo de modificaciones revisado con el impacto de la solución implementada	55

Capítulo 1

Introducción

En este capítulo se presenta el contexto necesario para comprender la situación actual y la problemática a abordar. Para facilitar el entendimiento de los términos propios del negocio, se provee un glosario (sección 10) con una breve descripción de cada uno.

1.1. Contexto

Este trabajo de memoria se enmarca en la operatoria propia de la empresa LeanSight¹, una consultora en innovación y tecnología conformada por ocho personas, especialistas en diferentes áreas (incluyendo Pedagogía e Ingeniería de Software), quienes colaboran en torno a la conformación de material didáctico y al dictado de talleres educativos y cursos de formación para profesionales. El propósito de la compañía es el siguiente:

“El mundo del trabajo está cambiando continuamente y cada vez de forma más acelerada. Las organizaciones están viendo cuestionado su paradigma de funcionamiento y se está haciendo necesario el adquirir la capacidad de prosperar humanamente en la incertidumbre, lo que llamamos Agilidad Esencial.

Nuestro aporte consiste en reorganizar las dinámicas humanas dentro de las organizaciones, para así liberar el talento de las personas. Esto, a través de habilitar el aprender, explorar, decidir y crear valor colaborativamente“²

LeanSight ofrece talleres orientados a la mejora de la cultura empresarial, organizados en sesiones de trabajo que están alineadas a aprendizajes específicos esperados. Un conjunto de contenidos coherente, que típicamente se aborda en media jornada de trabajo, es llamado módulo educativo, y su estructura será detallada más adelante en este documento.

Las ocho personas que conforman el equipo de trabajo colaboran con el objetivo “apoyar a las personas para que trabajen mejor”, encargándose de que aprendan de la mejor manera posible. Este alineamiento requiere que la formación que entregan sea adaptada para cada

¹<http://www.leansight.com>

²Cita de Agustín Villena, fundador de LeanSight

cliente, y vaya mejorando en el tiempo de la forma más ligera posible.

LeanSight cuenta con dos personas a cargo de la parte digital de la empresa, ambos desarrollando actualmente su respectivo trabajo de título para optar a Ingeniería Civil en Computación (y uno de ellos el autor del presente documento). El área digital tiene como objetivo cubrir las necesidades tecnológicas de la empresa, en cuanto a automatización, modelamiento, mantenimiento y exploración de nuevas tecnologías.

Tras el reciente paso a los editores de Google (Google Docs y Google Slides), el área digital ha empezado a desarrollar extensiones a sus funcionalidades para simplificar los procesos mecánicos durante la modificación de los recursos educativos que apoyan sus talleres. Todo esto se realiza a través de Google Apps Scripts, un servicio de Google que será explicado más adelante.

1.2. Problemática a abordar

Con el objetivo de mejorar continuamente la experiencia de aprendizaje, los talleres que LeanSight ofrece son constantemente modificados, en particular cada vez que se tenga una observación nueva, así como también después de cada instancia de feedback que obtengan de los alumnos o durante la revisión general de los módulos. Estas modificaciones tienden a ser muy pequeñas, y en principio, de fácil ejecución, tales como el cambio del orden de las diapositivas, reemplazo de imágenes o expresiones, o inserción de una diapositiva nueva.

A pesar de lo simple que parezca este tipo de modificaciones, el proceso de corrección de estos recursos y de sus vínculos con otros materiales educativos es tremendamente costoso. Al estar los distintos recursos fuertemente relacionados entre ellos, los cambios en uno impactarán rápidamente en el resto. Por ejemplo, mover un contenido dentro de un guion requiere modificar la presentación, el cuadernillo y la guía del facilitador. Incluso, después de tener esta precaución, habría que asegurarse que las referencias a páginas específicas (por ejemplo, desde la presentación al cuadernillo) mantengan su integridad. Es así como un cambio pequeño en este escenario se transforma en una reestructuración de todos los recursos.

Sin embargo, este problema tiene un alcance mayor; la modificación de contenidos requiere asegurar que los cambios se hayan propagado correctamente, abriendo mucho espacio para la introducción de errores humanos. Olvidar actualizar una referencia puede, como mínimo, confundir a los participantes de un taller y perturbar su proceso de aprendizaje. Incluso habiendo realizado todas las modificaciones pertinentes, puede ocurrir, por ejemplo, que involuntariamente se sobrescriban las modificaciones por desincronización, tanto dentro del equipo de trabajo como en la actualización de Google Drive, provocando incongruencias entre versiones.

Se ha intentado abordar la problemática de desincronización agregando manualmente etiquetas de versiones que permitan al facilitador detectar estas problemáticas oportunamente, pero escribir estas etiquetas aumenta la cantidad de tiempo necesario para cada modificación en el cuerpo de contenido, agravando el problema de la tediosa gestión de contenido. Es necesario estudiar esta necesidad en profundidad y automatizar los procesos que lo requieran.

Por otro lado, la baja modularidad de los componentes y la poca visibilidad de los aprendizajes que se persiguen, vuelven complejo el flujo de los talleres para personas que no sean expertas en la temática. De esta manera, solamente un par de personas (los expertos), pueden siempre realizar cambios en el cuerpo de contenido, pues tienen claridad sobre la función de cada elemento. De igual forma, solamente los expertos pueden crear nuevos módulos de aprendizaje o, como mínimo, deben supervisar el proceso.

Algunas actividades educativas tienen varios aprendizajes asociados, por lo tanto, éstas frecuentemente pertenecen a varios talleres a la vez. Después de modificar una de las actividades recurrentes, se deben modificar todas las ocurrencias de la actividad; esto es un trabajo mecánico muy costoso que requiere automatización, pues realizarlo a mano implica un alto riesgo de introducir errores o dejar contenido desactualizado. Esta falta de modularidad de los recursos educativos dificulta a la vez la construcción de nuevos talleres, donde hay que copiar y pegar contenido específico desde otros módulos. En la Figura 1.1 se muestra el diseño general de un taller en específico, haciendo referencia a componentes que ya han sido utilizados en otros talleres que podrán ser incorporados con ciertas modificaciones, brindando una idea de modularidad que podría ayudar a automatizar trabajos costosos para la empresa.

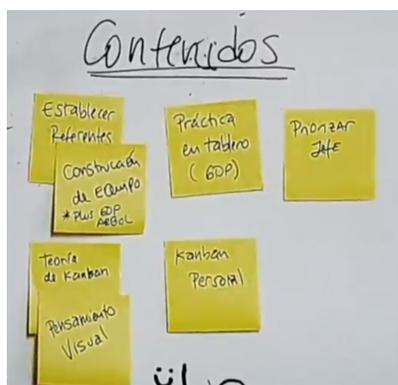


Figura 1.1: Ejemplo de diseño de módulo educativo

A pesar de que LeanSight tiene una estructura específica de módulo educativo y es resultado de varios años de experimentación, otras personas y organizaciones que requieran administrar contenido tienen problemáticas similares, como profesores, periodistas, escritores, estudiantes y otras consultoras. Sin embargo, estos problemas tienen características que los vuelven muy específicos y los hacen requerir probablemente una solución desarrollada a medida. LeanSight tiene un cuerpo de contenido que se compone de múltiples módulos, cada uno compuesto por varios documentos fuertemente relacionados entre sí, de manera que cualquier editor estándar en el área tendrá limitaciones naturales para resolver este problema.

El equipo frecuentemente se reúne con expertos de todo el mundo asistiendo a cursos y conferencias mundialmente reconocidos para mantenerse actualizados dentro de los tópicos que deben manejar. En las conversaciones con otros consultores que utilizan varios tipos de documentos a la vez, no se ha observado que hayan empezado a aplicar estrategias de mejora continua compatibles con el trabajo colaborativo sobre sus recursos, por lo que entregar la solución que este trabajo pretende le entregará a la organización una alta ventaja competitiva y los volvería referentes en el mundo.

Otras consultoras y en particular grandes coaches y formadores profesionales del mundo enfrentan la problemática de forma similar al lanzamiento de libros, donde la evolución involucra grandes cambios y son solo incorporados en nuevas ediciones del cuerpo de contenido, publicados con muy baja frecuencia, lo que produce rápidamente discrepancias entre conceptos que se relacionan, por actualizarse en períodos distintos. De cualquier modo, esto es esperable para consultores reconocidos, porque sus capacitaciones certifican conocimientos específicos, en torno a una terminología particular y muchas veces están justamente acompañadas de libros que comprenden los contenidos de la forma en que fueron adquiridos, dejando poco espacio para la mejora continua y a veces volviéndolos comprensiblemente inconsecuentes con los paradigmas que usan. Romper este paradigma y habilitar la focalización en la construcción de conocimiento y mejores prácticas educativas, aunque podría no ser siempre posible, traería enormes beneficios para cualquier organización en el área.

1.3. Situación actual

El foco de este trabajo se encuentra en la evolución de un cuerpo de contenidos educativos implementado en documentos virtuales, por lo tanto, la situación actual se describe mediante su estructura y su soporte tecnológico, que serán detallados a continuación.

1.3.1. Cuerpo de contenidos educativos

Las actividades educativas que LeanSight lleva a cabo a lo largo de los talleres se apoyan en varios recursos educacionales, cada uno con una función específica. A continuación, se indican los más importantes:

- *Presentación*: Este recurso sirve de apoyo visual al facilitador a cargo de las actividades y provee una guía general para los contenidos que se abordarán durante el taller.
- *Cuadernillo*: Es un documento impreso entregado a cada participante del taller, donde puede tomar notas y realizar las actividades que incluye. Se acostumbra a regalar una carpeta para que el participante pueda guardar todos los cuadernillos que reciba y los pueda revisar.
- *Guía del facilitador*: Esta es una guía que explica detalladamente lo que el facilitador debe hacer en cada sección, al nivel de que cada diapositiva de la presentación tiene anotaciones asociadas.
- *Otros recursos*: Este recurso corresponde a cualquier otro tipo de soporte audiovisual, tanto para la realización misma de los talleres, como para la administración de estos, típicamente son videos con montajes o charlas que sensibilicen al participante, buscando que descubra, desde la reflexión, la importancia de los objetivos del módulo.

Los recursos están fuertemente relacionados entre sí, pues corresponden al mismo hilo conductor del taller. La presentación, que actúa como el conductor visual, contiene referencias a páginas específicas del cuadernillo, que indican cuándo el participante debe realizar una actividad o iniciar una reflexión. También tiene vínculos a los videos que se deben reproducir en momentos determinados, y las instrucciones para realizar actividades en equipo. Por lo tanto, todo el proceso educativo de un taller se vuelve visible, permitiéndole al facilitador entregar los aprendizajes de forma efectiva.

La fuerte relación entre los recursos educativos proviene de una idea intangible del flujo que debe seguir el taller. A este flujo le llamaremos *guion* o *relato*, que corresponde a la estructura lógica de los talleres y consiste en un conjunto de contenidos a entregar, actividades a realizar y consideraciones a tener.

La unión de estos recursos interrelacionados en torno a una sesión o un conjunto de sesiones de trabajo es actualmente llamada un *módulo educativo*. Estos módulos están presuntamente compuestos por elementos indivisibles llamados *componentes*. La exploración y el descubrimiento de dichos componentes se encuentra entre los objetivos de este trabajo.

1.3.2. Soporte tecnológico

LeanSight cuenta con el servicio G Suite de Google, que facilita la integración de los quehaceres de la organización, con los diversos servicios que provee Google. De esta manera, todos los archivos relevantes son administrados a través de un servicio basado en Google Drive (cloud service). Los módulos educativos están separados en documentos independientes y la Figura 1.2 muestra cómo cada uno está almacenado en su propia carpeta.



Figura 1.2: Algunos módulos educativos almacenados en Google Drive

El almacenamiento y la edición de los documentos del cuerpo del contenido han pasado por varias plataformas, siendo las más estables las de Microsoft Office y Google Editors. Específicamente se utiliza Microsoft Word para los cuadernillos y Microsoft PowerPoint para editar el apoyo visual de los talleres. Al encontrar problemáticas en el trabajo colaborativo se realizó hace poco un traspaso de todos los documentos de PowerPoint a Google Slides, que evita problemas graves de sincronización y se habilita el trabajo en paralelo sobre el mismo documento. Se considera actualmente pasar también los cuadernillos a Google Docs, si sus funcionalidades lo permiten. De esta manera se puede empezar a utilizar Google Apps Scripts (GAS)³, un servicio de scripts de Google basado en Javascript, que mediante una API

³<https://developers.google.com/apps-script/>

brinda acceso tanto a los archivos y carpetas de Google Drive, como al propio contenido de los documentos, extendiendo las funcionalidades de los editores de Google.

El desarrollo de Google Apps Scripts, además de las complicaciones propias derivadas del desarrollo en Javascript, es muy incómodo para extender el editor provisto por Google, ya que impide cualquier intento de control de versiones o soporte al trabajo colaborativo. Esta problemática se logra cubrir con clasp⁴, una herramienta escrita en Node.js⁵ y distribuida por npm⁶ que no solo permite el desarrollo local y el uso de control de versiones (como Git⁷), sino también el uso de Typescript⁸. Este código es automáticamente traducido a Javascript al subirlo a Google mediante un simple comando, brindando así extensibilidad y mantenibilidad mediante el uso de clases y anotaciones de tipo, que requeriría mucho tiempo de desarrollo si se quisiera implementar en Javascript.

1.4. Objetivos de la memoria

Las problemáticas antes planteadas se traducen en: (1) una falta de visibilidad en los aprendizajes perseguidos por cada elemento de los talleres, (2) poca claridad sobre la función y los requisitos de cada actividad, (3) procesos de edición de documentos altamente tediosos y costosos, (4) dependencia de expertos para el modelado de talleres, y (5) limitaciones para incorporar nuevos facilitadores o diseñadores educacionales. Por lo tanto, este trabajo de memoria busca diseñar e implementar una herramienta a medida, que facilite estos procesos y mejore la visibilidad de todos los aspectos necesarios para crear, diseñar y realizar cada actividad educativa asociada con los talleres que imparte LeanSight.

El objetivo general de este trabajo de título es desarrollar una herramienta o un conjunto de herramientas de software que asistan el proceso de construcción de nuevos talleres y la mantención de los ya existentes. La herramienta deberá ser consistente con la estructura de los modelos definidos por la empresa, y usable para cualquier miembro de LeanSight. Los objetivos específicos que se derivan del objetivo general son los siguientes:

- Definir un formato de representación de módulos educativos que sea flexible y modular, orientado a la fácil modificación y reutilización.
- Desarrollar un conjunto de servicios de software que permitan al equipo la modularización y reutilización de componentes educativos.
- Desarrollar un componente de software que permita simplificar la creación y evolución del cuerpo de contenidos educativos de LeanSight.
- Desarrollar la herramienta que integre los componentes anteriores y permita alcanzar el objetivo general antes definido.

⁴<https://developers.google.com/apps-script/guides/clasp>

⁵<https://nodejs.org/en/>

⁶<https://www.npmjs.com/>

⁷<https://git-scm.com/>

⁸<https://www.typescriptlang.org/>

1.5. Estructura del documento de memoria

El documento de memoria detalla el trabajo realizado en siete capítulos, buscando comprender la problemática y resolverla efectivamente. Primero se estudia la problemática incorporándose al equipo y viviendo sus quehaceres (sección 2), luego se modelan los flujos de trabajo a partir de las observaciones obtenidas (sección 3). Habiendo alcanzado un mejor entendimiento del problema a resolver, se reflexiona en busca de un desafío focal en el capítulo de Análisis de la situación actual (sección 4). El diseño y concepción de la solución se explica en su propio capítulo, fundamentando las decisiones sobre elección de framework y modelo de datos (sección 5). La implementación de la plataforma se divide por funcionalidad, explicando brevemente las decisiones relevantes y la forma en que apunta al desafío focal planteado (sección 6). Con la plataforma implementada, se valida si está solucionando la problemática a través de métricas de impacto y un caso práctico (sección 7). Finalmente se documentan conclusiones del trabajo y funcionalidades futuras que mejorarían la plataforma, ayudando a cubrir de mejor manera las problemáticas descubiertas y apoyándose en la investigación realizada.

A pesar de que esta documentación sigue la estructura descrita, el trabajo no se realizó de forma secuencial, sino que se siguieron ciclos cortos con validaciones frecuentes. El desarrollo basado en experimentos cortos garantiza que el camino que se sigue es el correcto y las validaciones frecuentes proveen retroalimentación oportuna. La estructura del documento está pensada para entregar al lector todas las observaciones y conclusiones relevantes de forma clara y se debe tener presente que tal documentación se confecciona a posteriori.

Capítulo 2

Estudio de la problemática

En este capítulo se describe un estudio inicial de la problemática, que servirá como base para crear modelos y realizar observaciones que permitan visualizar los síntomas relevantes de forma clara y accionable. Se comienza con una inmersión inicial en los quehaceres del equipo, el marco de trabajo pedagógico con que se diseña contenido y se estudia la forma en que se socializa el conocimiento dentro del equipo, con un modelo concreto que podría entregar mejoras tempranas a los flujos de trabajo. Además, se exponen las soluciones pequeñas de automatización o gestión que se diseñaron e implementaron durante el proceso.

2.1. Viviendo el proceso

El trabajo comienza incorporándose al equipo y viviendo cada uno de sus procesos, bajo el principio *genba* (現場), que significa “el lugar actual” o “escena del crimen” en otros contextos. El término *genba* se refiere al lugar en que el trabajo ocurre y es parte fundamental de la filosofía de Toyota, donde se realizan *genba walks*, en las que se transparenta el trabajo de las áreas donde se supone una potencial mejora [4]. De esta forma, se apoya activamente el principio *kaizen* (改善), en español “mejora continua”, mediante una plena comprensión de las problemáticas cotidianas.

Durante el proceso de involucramiento en los quehaceres de *LeanSight*, surgieron varias ideas de solución que fueron rápidamente implementadas en sus versiones más simples y sobre las que se sigue iterando según las prioridades del equipo.

2.1.1. Preparación de talleres

La experiencia *genba* comienza con la preparación de un taller. Primero se preparan todos los recursos digitales necesarios. En el momento en que esta experiencia comenzó, todos los documentos se editaban en un ambiente local y se sincronizaban con Google Drive; entonces era necesario verificar que todos los documentos estén efectivamente actualizados según los cambios que haya realizado algún diseñador educacional.

Para solucionar esta problemática, se evaluaron las funcionalidades que ofrecen Microsoft

Office y los editores de Google. Se verificó que no eran estrictamente necesarias las funcionalidades de Office y se decidió usar Google Docs y Google Slides para la gestión del contenido, pues facilita el trabajo colaborativo y da la posibilidad de programar extensiones que automaticen tareas tediosas.

Después de incorporar esta mejora, en talleres posteriores, se observa que persiste la problemática del escaso control sobre los cambios realizados y no se tiene seguridad sobre la integridad de los documentos ni se puede acceder a versiones específicas de documentos, para compartirlas con los participantes de un taller o utilizarlas de base para otra versión. Aquí se inicia en concreto el proyecto de control de versiones para los documentos de Google, del que se hablará más adelante en este capítulo.

Teniendo ya los recursos digitales recopilados, se deben guardar todos en un pendrive de respaldo, en caso de que falle el computador que portará la presentación. Además, se deben imprimir los cuadernillos en un centro de impresión, para luego compaginarlos en carpetas según la cantidad de participantes esperados, siempre con un margen de seguridad de unas pocas carpetas adicionales. Los clientes acostumbran a confirmar la cantidad de asistentes uno o dos días antes del taller y esto implica un conjunto de problemáticas de logística, práctica que es necesario cambiar.

Los talleres contienen actividades prácticas, cada una con materiales específicos. Existe lista de materiales que se debe llevar en general, pero es poco específica y simplemente nombra la actividad, asumiendo que quien prepara el taller sabrá en detalle lo que significa. Se modificaron las listas cada vez que el nivel de detalle pareció insuficiente. Además, se identifica redundancia en la información, porque la mayoría de los elementos son requeridos para cualquier taller y solo unos pocos son específicos de los contenidos y actividades correspondientes.

El proceso general de preparación toma tiempo y es tedioso, y cuando se debe hacer tan rápido como sea posible, se vuelve particularmente problemático. Antes de imprimir uno de los talleres, se encontró un error importante en los documentos, y se tuvo que volver a pasar por varios de los pasos que aseguran la concordancia de los documentos, pero esta vez con mucha prisa, abriendo espacio para la aparición de nuevos errores.

2.1.2. Realización de talleres

Con todos los materiales preparados, se puede iniciar la sesión de trabajo en las instalaciones que el cliente provea. El equipo de LeanSight debe llegar con anticipación para asegurar que cumple con las necesidades técnicas, esto es, mesas y sillas que se puedan mover para realizar las actividades necesarias, televisión o proyector para mostrar el apoyo visual del facilitador y otras necesidades que pueden variar de taller a taller.

Las sesiones son típicamente facilitadas por uno o dos miembros de LeanSight. Desde la primera experiencia se observa que el facilitador debe tener amplio conocimiento sobre el taller que realiza, conocimiento que no se encuentra explícito en ningún tipo de registro. De esta manera, el experto parece ser el único que puede, en principio, realizar un taller y el resto del equipo solamente puede aprender observando la ejecución e incorporándose gradualmente, transmitiendo el conocimiento a través de la experiencia compartida. Este

tema se profundizará en la sección 2.2.

En cierta oportunidad, una lista de conceptos específica, propia de una actividad, estaba solamente en la memoria de un miembro que no se encontraba la sesión en que se requeriría; este problema se podía predecir, pero la baja visibilidad de los contenidos que se trabajan en la sesión no lo permiten. El problema no debería volver a ocurrir después de almacenar la lista en un medio adecuado, pero la conclusión relevante se encuentra en una visibilidad deficiente de lo que el facilitador requerirá para la sesión, que podría causar nuevas problemáticas.

Durante una de las sesiones, uno de los facilitadores, buscando que los aprendizajes se adopten de mejor manera y basado en su extensa experiencia, ideó un cambio en el orden de dos contenidos; esto implicaba además agregar un video y un par de diapositivas desde otro módulo, para mantener un flujo natural entre conceptos. El cambio resultó efectivo en cuanto a aprendizajes, pero dejó inmediatamente obsoletos los cuadernillos, y no solo dejó a los participantes confundidos al ver que los documentos no se correspondían con ese cambio, sino que también incomodó al otro facilitador, que tenía menos experiencia con esos contenidos. Aquí se evidencia de nuevo la fuerte relación entre los recursos educativos, pero además se conversó que puede ser buena idea implementar algún tipo de protocolo, que prohíba realizar ese tipo de cambios dentro de la misma sesión.

Al final del taller se hace una actividad de evaluación por parte de los participantes, es importante que sea muy breve y no les agote, así que simplemente se les pide escribir en un papel una nota de 0 a 10 y comentarios generales que quieran compartir. Los resultados se usan calculando el Net Promoter Score (NPS)¹, pero siempre se consideran nuevas formas de evaluar según el estado del conocimiento. Los comentarios que los participantes dejen se llevan a una instancia de retrospectiva, de donde habitualmente nacen múltiples cambios por realizar sobre los módulos educativos.

Después de realizar la sesión, se comparte a los participantes los materiales audiovisuales usados. Por el lado de los facilitadores, esto implica buscar cada uno de los documentos utilizados y exportarlos, pero como el contenido se mantiene en constante cambio, los documentos podrían ya haber cambiado e incluso ser incompatibles con la experiencia de los participantes. Actualmente se recurre a la práctica de exportar los archivos para compartir durante la preparación del taller, solucionando el problema, pero lo interesante es el problema de baja visibilidad de los cambios realizados sobre el cuerpo de contenidos, que puede provocar varias otras adversidades y en particular el problema de creación de módulos, que se discutirá en la sección 2.1.3.

2.1.3. Modificación de talleres

La modificación de talleres y la sincronización de sus cambios en base a editores de Microsoft Office y la nube de Google Drive, es altamente caótica cuando se quiere habilitar el trabajo colaborativo y el acceso compartido. El equipo ha estado consciente de esta problemática por mucho tiempo y ha experimentado algunos espacios de solución. Para abordar la problemática, actualmente se utiliza una hoja de cálculo de Google Sheets, que mantiene información sobre la estructura de los talleres en cuanto a aprendizajes esperados y activida-

¹<https://www.netpromoter.com/know/>

des educativas realizadas, incluyendo también cada evento en el apoyo visual y el cuadernillo del participante. En paralelo se mantiene un tablero de Trello² llamado “Issue Tracker”, con el que se mantiene seguimiento de todos los cambios hechos y por hacer.

Entre los dos registros, hay mucha información redundante y difícil de mantener, haciendo más costoso el proceso de cambio, pero siendo necesarios para poder visibilizar la realización efectiva de modificaciones. Aquí se inició un proyecto pequeño en que se conectan ambas plataformas, de manera que el marcado de una tarea como “realizada” o la creación de una nueva tarjeta, se puedan realizar desde la misma hoja de cálculo, sin necesitar entrar a Trello y facilitando un mejor seguimiento de cambios.

La primera problemática que se evidencia durante la modificación misma de un taller está en que son muchos documentos que deben ser visualizados de manera simultánea. Como ya se ha discutido, los documentos no son independientes entre sí, y casi en ningún caso es razonable editar uno de los recursos sin considerar el resto.

Después de realizar una modificación concreta en uno de los recursos que conforman un módulo educativo, no se tiene certeza de cuán lejos el cambio se debe propagar, podría ser necesario cambiar todas las referencias de la presentación al cuadernillo, o incluso tener que modificar otro módulo en que se utilizó el mismo recurso. Estos problemas surgen naturalmente de que el soporte tecnológico no sea coherente con la lógica de los documentos. Los recursos educativos están fuertemente relacionados en toda su extensión, y el hecho de considerarlos independientes, vuelve cualquier modificación necesariamente compleja y tediosa.

2.1.4. Confección de talleres y módulos

La confección de un taller muchas veces carece de un flujo de trabajo claro, y es parte de los procesos que a LeanSight le interesa estudiar en profundidad. Este tema es presentado más en detalle en la sección 3.

A grandes rasgos, la construcción de un nuevo taller empieza con la selección de contenidos ya desarrollados en base a las necesidades negociadas. Rara vez se tiene acceso a las versiones específicas que se deben usar o a veces es muy complicado encontrar documentos con la información relevante.

En concreto, se observa que no se puede acceder a versiones que fueron utilizadas en algún taller, y la situación es particularmente compleja cuando se hacen modificaciones específicas para cierto cliente, porque actualmente se sobrescribe la versión “oficial” con versiones específicas.

En cualquier caso, los miembros especializados en educación deben asegurar que las actividades educativas y metacognitivas sean apropiadas y cumplan con sus objetivos. Se entiende por actividad metacognitiva como un tipo de evaluación en la que se asegura la incorporación de los aprendizajes esperados y se realizan al final de cada módulo o sección.

Además de todo lo anterior, podría requerirse un módulo educativo completamente nuevo, incluso sin un cliente que lo requiera explícitamente. Este proceso requiere de todas formas

²<https://trello.com>

reutilizar ciertas secciones de módulos ya existentes, porque cada experiencia tiene ciertos aprendizajes como requisito y el flujo del taller debe ser lo más natural y comprensible posible.

2.2. Haciendo explícito el conocimiento del equipo

El equipo de LeanSight se conforma con personas de distintas especialidades y todos trabajan en conjunto para (1) diseñar propuestas para clientes, (2) diseñar contenido orientado al correcto aprendizaje con bases pedagógicas, (3) confeccionar contenido educativo, (4) modificar el cuerpo de contenido a partir de observaciones en talleres y (5) cumplir el rol de facilitador en la realización de talleres.

A pesar de que el equipo tiene profesionales de diversas áreas, todos desempeñan, a grandes rasgos, las mismas actividades, y colaboran en torno al objetivo de generar una buena experiencia de aprendizaje, que apoye a las personas a mejorar su forma de trabajo. Aun así, inevitablemente se alcanzan distintos niveles de entendimiento en el equipo; algunos se sienten más cómodos con actividades y contenidos específicos, dándonos una pista de que el entendimiento podría no estar siendo compartido de forma efectiva.

El proceso en que el conocimiento se comparte y se incorpora ocurre tradicionalmente mediante la vivencia de experiencias en grupo; es decir, haciendo lo que se quiere aprender, en conjunto con quienes ya lo conocen bien. El aprendiz en este caso recibe el nombre de “ayudante de mago”, asumiendo un rol con responsabilidades crecientes a medida que gana conocimiento mediante la experiencia.

Luego de algunas conversaciones con el equipo y una investigación posterior, se descubre que un buen modelo para este proceso de socialización del conocimiento es el reconocido modelo SECI [5]. Este modelo describe un proceso de construcción de conocimiento, alternando entre conocimiento explícito y tácito mediante cuatro acciones clave:

- *Socialización* (tácito a tácito): Se genera nuevo conocimiento a partir de interacción social dentro de la cotidianidad.
- *Externalización* (tácito a explícito): El conocimiento tácito se articula para volverlo explícito, haciendo posible compartirlo con otros, habilitando la generación de nuevo conocimiento.
- *Combinación* (explícito a explícito): Proceso de recolección y combinación con conocimiento desde otras fuentes, volviéndolo explícito de forma sistemática.
- *Internalización* (explícito a tácito): Dentro de la organización se comparte el conocimiento y cada individuo lo internaliza, haciéndolo parte de la rutina diaria.

Este modelo representa un fundamento de la metodología de este trabajo de título y exponer este modelo al equipo habilita la construcción de conocimiento, articulando y cuestionando hábitos y creencias, para eliminar cualquier elemento que pueda entorpecer la mejora continua.

Algunos problemas que puede evitar este modelo son: expertos que creen que todos manejan el mismo nivel de conocimiento que ellos, especialistas que no comunican efectivamente sus ideas y aprendices adoptando malas prácticas en virtud de la tradición.

Construir conocimiento sin volverlo explícito hará que solamente los expertos puedan participar efectivamente de los quehaceres de LeanSight, poniendo una gran barrera para nuevos facilitadores o diseñadores educacionales. La siguiente sección representa una aplicación directa de este modelo en las retrospectivas de cada taller.

2.3. Buenas prácticas pedagógicas

Las retrospectivas, que ocurren posteriores a cada taller, tienden a reformular muchas de las prácticas de LeanSight, no solamente la estructura de sus talleres. Un desafío que no se ha abordado hasta ahora está en mantener esas conclusiones visibles para todo el equipo y buscar que se incorporen efectivamente a la cultura de la organización.

Las conclusiones más difusas son del tipo “Poner énfasis en cierto principio” o “Interactuar con los participantes de esta forma en torno a este tema”; otras son más concretas como “imprimir cierto material para cierta actividad” y todas estas conclusiones están siendo consideradas como cambios distintos a la modificación de los recursos educativos mismos.

Tomando todas las consideraciones que se debían tener durante los talleres de LeanSight, se analizaron alternativas para almacenarlas. Un primer acercamiento consiste en utilizar las notas del orador en la presentación de Google Slides, pero algunas anotaciones tendrían que estar escritas en todas las diapositivas, por ser consideraciones generales para el taller, y otras debían estar antes de la primera positiva, por referirse a la preparación del taller o el inicio de este.

Se observó que las consideraciones problemáticas no podrían ser escritas de ninguna forma en la presentación misma, porque no pertenecen al mismo nivel de especificidad. De esta forma, se separaron en tres niveles con cierta jerarquía:

- *Prácticas de LeanSight*: Son buenas prácticas generales, que son independientes de los contenidos y aprendizajes que se estén entregando, relacionadas principalmente a la pertinencia de materiales y buenas relaciones con los participantes.
- *Consideraciones para el módulo*: Son propias de los contenidos que se abordan. Tienen que ver con enfatizar términos clave y con involucrar a los participantes de forma adecuada.
- *Detalle de actividades educativas*: Son consideraciones en torno a actividades concretas. Este es justamente el tipo de anotaciones que se pueden mantener en las notas del orador, pues están contenidas en el período en que se muestra una única diapositiva o los participantes realizan una actividad en equipo.

Este modelo muestra con claridad por qué no puede anotarse todo en un mismo lugar y se deben llevar a varios recursos. Actualmente se utilizan un par de listas para asegurar que el facilitador recuerde todos los materiales que debe llevar a cada taller, una lista global, que se aplica para cualquier taller, y una lista específica de cada módulo educativo. Esta lista actualmente no se imprime, sin embargo, en una retrospectiva se descubrió la importancia de hacerlo, así que, junto con instaurar esta práctica, se implementaron las observaciones en las mismas listas según corresponda.

2.4. Control de versiones

Varias de las problemáticas ya descubiertas se pueden englobar en el tema de control de versiones, es decir, la posibilidad de rastrear cambios y asegurar que los documentos están en la versión más actualizada, pero entendiendo que esto no corresponde a una causa raíz, sino únicamente a una categoría. La necesidad de visibilizar la integridad de los documentos se vuelve crecientemente importante, existen cada vez más versiones similares en paralelo y los facilitadores comienzan a confundirse en el estado de cada módulo.

Este tema fue desarrollado en conjunto con Lucas Puebla, estudiante de Ingeniería Civil en Computación, y de a poco se ha vuelto el foco principal de su trabajo de título, que durante el desarrollo de este documento se encontraba en proceso, y terminado a la fecha. En dicho documento se puede encontrar este desarrollo en detalle [2].

Frente a los problemas de desincronización y descoordinación en torno a los recursos y sus versiones, se adoptó recientemente la práctica de escribir etiquetas de versiones cada vez que un recurso se edita. Si bien esto elimina el problema por completo, aumenta en gran medida el tiempo necesario para realizar cambios; una persona con poco manejo de tecnología podía tardar fácilmente media hora en escribir la versión en el contenido de cada documento y el nombre de cada archivo exportado. Considerando la alta frecuencia en que se espera realizar cambios, se requiere automatización dentro de este proceso.

El proyecto que aborda estas problemáticas, el *Sistema de Control de Versiones* de Lean-Sight, está implementado en Google Apps Scripts, que permite gestionar el contenido de documentos y su exportado programáticamente. El desarrollo está enfocado en mantener buena calidad de código, para habilitar su ágil evolución. Este proyecto tiene hasta la fecha las siguientes funcionalidades, que fueron desarrolladas en incrementos pequeños de acuerdo con las prioridades del equipo:

- Exportar automáticamente documentos de Google Docs en formato pdf (Portable Document Format, en español: Formato de Documento Portátil) y documentos de Google Slides en formato pptx (presentaciones de Microsoft PowerPoint) y pdf.
- Actualizar etiquetas en el cuerpo de los documentos, para escribir etiquetas de versión u otra información automatizable que se refleje en la impresión o proyección de los recursos.
- Archivos de metadatos para cada módulo, que entregan flexibilidad adicional a la de Google Drive en su manejo.

2.5. Revisión de sesiones y módulos

Explorando los archivos almacenados en Google Drive, se descubrió que hay gran cantidad de material duplicado; consultando con el equipo se aprendió que ésta es simplemente una implementación precaria para guardar las versiones de conjuntos de módulos que fueron utilizadas con clientes específicos, y que existe un registro mucho más grande almacenado en respaldos.

Se tiene un grupo pequeño de módulos almacenados en su propio ambiente, llamados Mó-

dulos Oficiales, que corresponden a los recursos que se usan como base con mayor frecuencia, varias veces sin modificar. Se comprendió que, fuera de los módulos oficiales, el equipo no hace diferencia entre los módulos editados para talleres específicos y talleres en desarrollo, haciendo necesarias nuevas denominaciones para esta implementación.

Se le dio el nombre Sesión a un conjunto de módulos utilizados en una fecha y un cliente específicos; se conforma de elementos que pueden provenir de módulos oficiales o de sesiones anteriormente realizadas, y pueden haber sido modificados con cambios específicos para la ocasión. Los módulos que se consideren incompletos son aislados de las dos categorías anteriores, en lo que se llama Laboratorio de nuevos módulos.

Estos conceptos son parte fundamental de la confección y modificación de módulos, así que se tendrán en consideración durante el resto del documento. Además, hacen necesaria la adaptación del modelo de control de versiones frente a la identificación de nuevos requerimientos. A esta nueva aplicación se le llama Navegador de Módulos y Sesiones, se apoya en el modelo de la sección 3.3 y tiene las siguientes funcionalidades:

- Una interfaz que permite seleccionar módulos desde las categorías de módulos oficiales, de laboratorio o usados en sesiones anteriores, para ser copiarlos automáticamente y usarlos de base para una nueva sesión.
- Nombrar nuevas sesiones y almacenar datos relevantes para su fácil identificación.
- Una interfaz que permita explorar cómodamente cualquier módulo (oficial, modificado o de laboratorio) y acceder a los contenidos de sus recursos.
- Mantener todos los módulos en listas que permitan automatizar su administración y eviten errores básicos de manejo.

Capítulo 3

Flujos de trabajo

Con la experiencia anterior, se alcanza un mejor entendimiento de la situación actual de la problemática. Sin embargo, para poder hacer concretos los síntomas identificados, es necesario visualizar los actores y recursos relevantes, además de los procesos que realizan. Este capítulo entregará los modelos necesarios y concluirá las problemáticas fundamentales de LeanSight en torno a la modificación y creación de contenido.

3.1. Actores y entidades

Dentro de LeanSight, existen tres roles fundamentales: (1) el *facilitador*, que se encarga de la realización de las sesiones y la retrospectiva posterior; (2) el *diseñador educacional*, que se encarga de la confección y mantención de módulos, y (3) el *experto*, que posee un amplio manejo de los contenidos y supervisa los hitos críticos de los talleres. Estos roles no son excluyentes entre sí y en general son compartidos por todos los miembros de LeanSight, esta es solo una representación de las distintas responsabilidades identificadas. Estos roles son implícitos dentro del equipo, y sus distinciones son parte de este trabajo.

Por parte del cliente, se pueden distinguir dos tipos de actores: (1) los *participantes*, que asisten a la sesión y se espera logren los aprendizajes pretendidos por el facilitador, y (2) el *contratante*, que tiene junto al experto las conversaciones relevantes para determinar los contenidos a abordar y los detalles del contrato.

Dentro de los recursos utilizados, se hace una distinción entre los *recursos educativos*, que participan de la entrega de aprendizajes, y los *recursos administrativos*, que visibilizan consideraciones generales dentro del equipo o el flujo de los talleres, como las checklists expuestas en la sección 2.3.

Estudiando los modelos que se han implementado, se identifica la existencia de un *guion* subyacente a cada módulo educativo, que representa el flujo general de los talleres. La idea de guion es parte fundamental de este estudio y será replanteado con frecuencia en este documento. Este flujo se puede visibilizar con un recurso administrativo concreto y ha pasado por varias implementaciones.

Los módulos educativos se conforman de actividades educativas, que incluyen cualquier evento del taller del que participan los asistentes, y persiguen cambios de paradigma articulados como *aprendizajes esperados*. Los aprendizajes esperados se alcanzan a través de experiencias que permitan al participante comprender por sí mismo la importancia y el efecto de adoptar un paradigma específico; distinción que se entiende como *quiebre* y algunos de ellos no pueden ocurrir sin haber adquirido sus respectivos *aprendizajes requeridos*.

El diagrama en la Figura 3.1 ilustra la relación entre estos actores y entidades. El contratante, usualmente con un equipo de trabajo, identifica necesidades entre sus equipos y en conjunto con un experto de LeanSight determina la necesidad presente, que se traduce en aprendizajes esperados. El experto selecciona actividades que producirán los quiebres necesarios, y el diseñador educacional arma un guion y prepara los recursos educativos necesarios. Al final de cada sesión se tienen *observaciones* del facilitador y *retroalimentación* de los participantes, que entregan una perspectiva nueva sobre el estado del módulo y las mejoras que requiere.

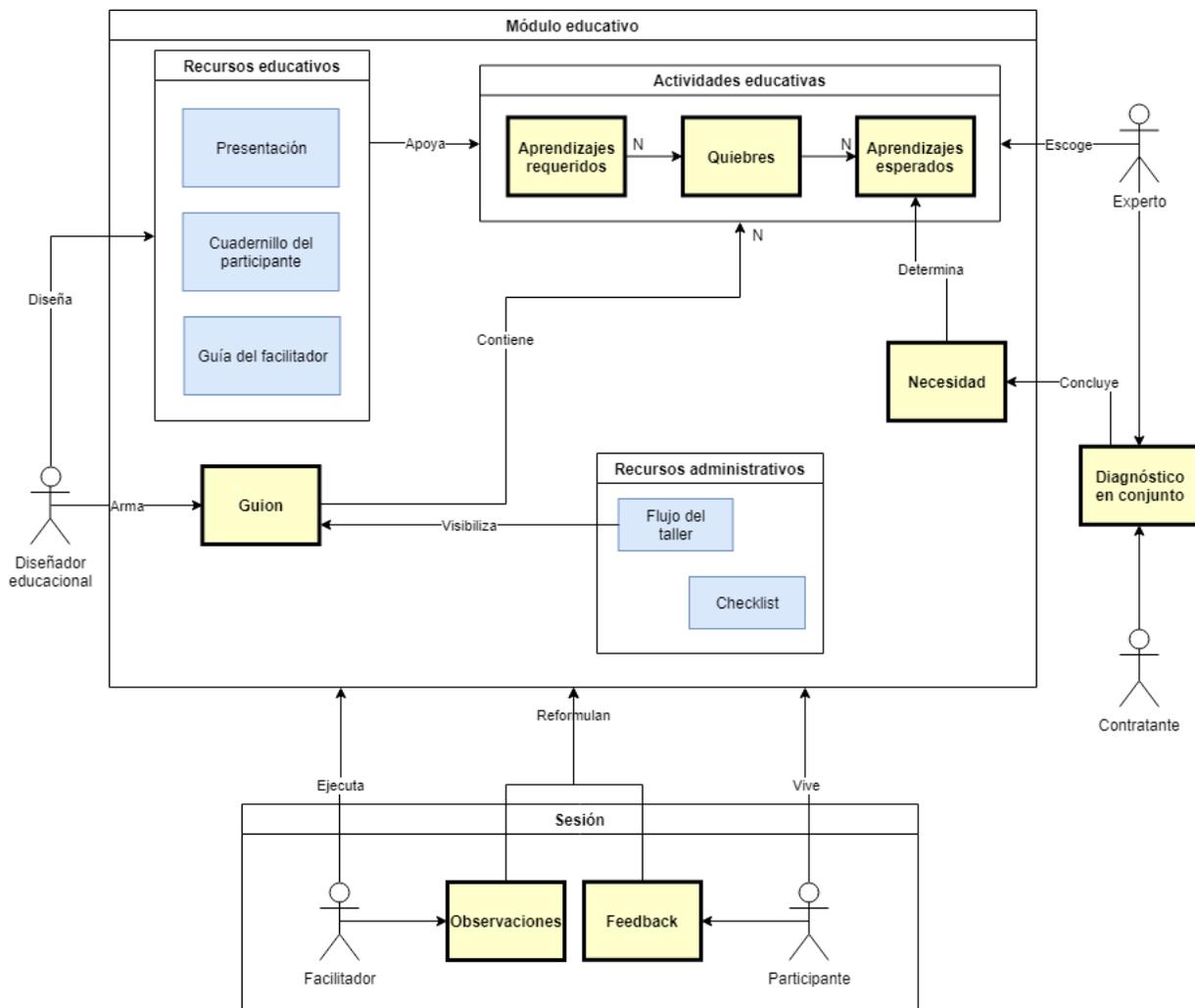


Figura 3.1: Diagrama de actores y entidades relevantes

Con lo anterior se modela adecuadamente la relación entre las entidades y da una idea de flujo de trabajo; sin embargo, el modelo para la selección de actividades educativas y la

confección de recursos educativos no es realista, porque esta información no se encuentra explícita para el equipo. Esto se condice naturalmente con el ciclo SECI, pues estas prácticas se basan, en un principio, en conocimiento implícito. Volver explícito este conocimiento será de gran importancia si el equipo quiere mejorar sus flujos de trabajo, porque lo volvería visible y legible para cualquiera que no sea un experto, permitiendo involucrarlos en todo el proceso.

3.2. Modificación de recursos

Ya se ha explicado lo rápido que evoluciona el contenido de LeanSight y algunos de los síntomas en torno a tal evolución. La Figura 3.2 muestra un modelo detallado del flujo de trabajo en la modificación de recursos según las observaciones realizadas durante el estudio de la problemática. En los tipos de modificaciones se enumeran solo algunas de las posibles, en particular las más frecuentes y concretas.

En la Figura 3.2 puede observarse cuán complejo puede volverse un cambio muy simple cuando requiere otros cambios, o se pierde la consistencia pedagógica. Tareas muy simples pueden tomar mucho tiempo por la arquitectura utilizada y algunas malas prácticas en el equipo relacionadas a herramientas de software incómodas o procesos tediosos. Dejar una tarea del flujo sin realizar puede tener un gran impacto en la siguiente modificación del módulo, por ejemplo, si la documentación no se actualiza, será muy difícil saber si el módulo está actualizado, o peor aún, saber si mantiene su integridad y consistencia pedagógica.

Por otra parte, el tiempo que requiere alguna tarea concreta no solo es elevado, sino que también incierto. Cambiar de posición un contenido no toma menos de media hora y puede fácilmente tomar una hora, porque los contenidos nunca están bien aislados, a pesar de que el equipo comprende que deberían estarlo. Reemplazar una imagen es particularmente difícil si no se conocen todos los lugares en que aparece, se ha experimentado en diez minutos y también en hasta dos horas, cuando es necesario revisar todo el módulo con todos sus recursos. Verificar la propagación de cambios por sí sola puede tomar veinte minutos y hasta una hora si el cambio era demasiado complejo o se hace sobre un concepto clave del módulo. Encontrar un módulo que refleje todos los cambios recientes es muy simple si se encuentra actualizado en la nube, pero si es necesario pedirles sus archivos a otros miembros del equipo, puede demorar más de dos horas e incluso no poder encontrarlo. Al quitar secciones del módulo para reducir su duración, a un experto no le toma menos de cuarenta minutos, tardando hasta una hora y media cuando no es una duración con la que se esté familiarizado. En cambios muy complejos la propagación de cambios es particularmente extensa; su sola verificación puede tomar hasta una hora.

La frecuencia con que se realiza cada tarea es relevante para comprender la magnitud del síntoma. Se realizan aproximadamente cuatro talleres por semana, debiendo pasar por el armado de sesiones (detallado en la siguiente sección) en cada uno. Los contenidos no se cambian de orden muy a menudo, pero la disminución de la duración del módulo se hace más de dos veces por semana. En total se hacen entre dos y tres cambios por semana y cada uno debe pasar por el flujo completo, donde podrían cometerse errores que vuelvan a las tareas futuras más difíciles de completar. La Tabla 3.1 muestra un resumen de la información antes presentada.

Tareas	Frecuencia (veces al mes)	Tiempo (min)
Cambiar de orden una actividad	2	30 - 60
Reemplazar una imagen	4	10 - 120
Quitar actividades para reducir su duración	10	40 - 150
Exportar recursos	15	15
Verificar la propagación de cambios	10	20 - 60
Encontrar un módulo actualizado	20	10 - 120

Tabla 3.1: Estimaciones de tiempo al ejecutar modificaciones típicas

3.3. Creación de sesiones

Como ya se ha expuesto, una sesión se compone de varios módulos que se esperan estudiar en media jornada o una jornada entera, que muchas veces deben ser modificados según el cliente. Este proceso cuenta con varios pasos, cada uno con sus propias problemáticas, que se ilustran en la Figura 3.3 y serán detallados a continuación:

1. *Selección de módulos*: Se seleccionan módulos con los que se quiera comenzar a trabajar, podrían requerirse módulos oficiales, o bien, módulos que ya hayan sido editados para sesiones anteriores. Este paso requiere encontrar versiones específicas de los módulos y es, actualmente, un proceso muy costoso, porque las sesiones se encuentran repartidas entre varios soportes y son difíciles de encontrar.
2. *Edición de recursos*: Los módulos seleccionados normalmente deben ser editados para la sesión que se está creando, es importante notar la cantidad de material duplicado que este paso podría generar y que en este paso no se tiene rastro de los cambios que ya han sido realizados, por lo que será necesario revisar los recursos completos con mucho detalle para saber si requieren algún cambio.
3. *Descarga de recursos*: Los recursos listos para la sesión deben siempre ser descargados, porque se considera riesgoso confiar en la conexión a internet del espacio entregado por el cliente. De este paso resultan numerosos documentos en formato pdf o pptx. Además, en este paso se debe verificar la coherencia de versiones, pues podría haber documentos desincronizados. Esta tarea nunca toma menos de 15 minutos en la situación actual y se hace cada vez más difícil a medida que se cometen errores en el proceso de modificación.
4. *Impresión*: Ciertos recursos deben ser impresos, como los cuadernillos o algunos materiales de actividades grupales. La principal complicación que esto conlleva es el compaginamiento de los módulos, que deben tener un orden específico dentro de la sesión.
5. *Ejecución de la sesión*: Con los recursos listos, se lleva a cabo la sesión. Hasta este paso, cualquier error cometido ya no es reparable y podría dañar enormemente la coherencia del taller.
6. *Adición al historial de sesiones*: La sesión ya fue realizada y debe ser almacenada para poder ser reutilizada, como en el paso 1. Es importante que la sesión se guarde con archivos en formatos editables y no como archivos publicados, es decir, se deben evitar documentos pdf, teniendo que volver al ambiente en que se editaron en el paso 2.
7. *Actualización del módulo oficial*: La adaptación de los módulos oficiales, después de

haberla puesto a prueba durante la sesión, podría querer hacerse definitiva. En este caso no es deseable el simple reemplazo del recurso, porque se perdería el rastro de los cambios ya realizados.

8. *Distribución de los recursos a los participantes*: Se debe entregar el material a los participantes para futuras referencias, de manera que se debe encontrar el material usado en la sesión y no se debe haber perdido por posibles cambios para otras sesiones.

3.4. Flujo creativo

El desarrollo creativo de talleres no se detiene nunca, es parte de la mejora continua y se espera que recupere retroalimentación después de cada sesión. El equipo ha ido mejorando su cuerpo de contenido por años, realizando experimentos pequeños en múltiples sesiones.

Un Diseñador Industrial está a cargo de la línea gráfica; dibuja las ilustraciones necesarias y se encarga del aspecto visual en general. El hecho de que él sea el encargado no evita que todo el equipo sea partícipe del proceso, pues es parte de la filosofía de LeanSight que todas las perspectivas son valiosas.

Parte fundamental del diseño de los módulos es el aspecto pedagógico, que está a cargo de una profesora con amplia experiencia en docencia y confección de material de estudio para colegios. Ella entrega la perspectiva teórica, por ejemplo, para mostrar la necesidad de involucrar a los participantes o de realizar actividades que evalúen la adopción de los aprendizajes perseguidos.

El fundador de la consultora es quién más experiencia tiene con los contenidos que se ofrecen; naturalmente es él quien toma habitualmente el rol de experto durante las sesiones de trabajo, asegurando que se mantenga el foco en provocar quiebres efectivos en los participantes y adquieran los aprendizajes que se buscan.

3.5. Síntomas y dolores identificados

Habiendo pasado por la inmersión inicial en la situación actual, podemos complementar y hacer concretos los síntomas encontrados.

- Casi todas las modificaciones requieren propagación a otros recursos y en la mayoría de las veces se hace incorrectamente.
- La propagación de cambios es manual, requiriendo mucho tiempo de trabajo en cada modificación.
- Se vuelven a armar módulos que ya se habían armado antes, incurriendo en mucho retrabajo.
- Las modificaciones están frecuentemente desconectadas con el objetivo de aprendizaje que se persigue.
- Hay patrones implícitos detrás del relato que le dan sentido desde una perspectiva pedagógica, pero no se mantiene íntegro por las frecuentes modificaciones incorrectas.
- La falta de patrones en el contenido hace difícil adaptarlo a la duración limitada de los talleres y formaciones, pues no es obvio qué partes se pueden recortar.

- La documentación de cualquier tipo requiere trabajo adicional y es difícil mantenerla precisa, útil y actualizada.
- No es clara la traducción entre los distintos tipos de recursos, es decir, no se puede comprender lo que debería verse en un recurso a partir de lo que se ve en otro.
- Aplicar un cambio simple a un recurso, toma 30 minutos en un caso optimista en la situación actual y puede fácilmente extenderse a 2 horas
- Preparar un taller nunca toma menos de medio día de trabajo de varias personas, incluso cuando el contenido original solo requiere modificaciones menores
- Los recursos sin clasificar aumentan constantemente, duplicando información tras la confección de toda sesión
- No se tiene seguimiento de los cambios realizados y puede tomar horas encontrar versiones específicas
- Gran parte de los cambios no se documentan de ninguna manera

Esta es una primera mirada a las problemáticas que ocurren en torno a la comprensión y edición de los recursos educativos. Sin embargo, identificarlas no es suficiente para alcanzar una solución, necesitamos primero estudiarlas y analizarlas en busca de una causa raíz, para solucionar el problema correcto. Se comenzará mirando en detalle las razones específicas por las que el cuerpo de contenidos evoluciona, esperando luego replantear la situación actual como una diferencia medible respecto a la situación deseada.

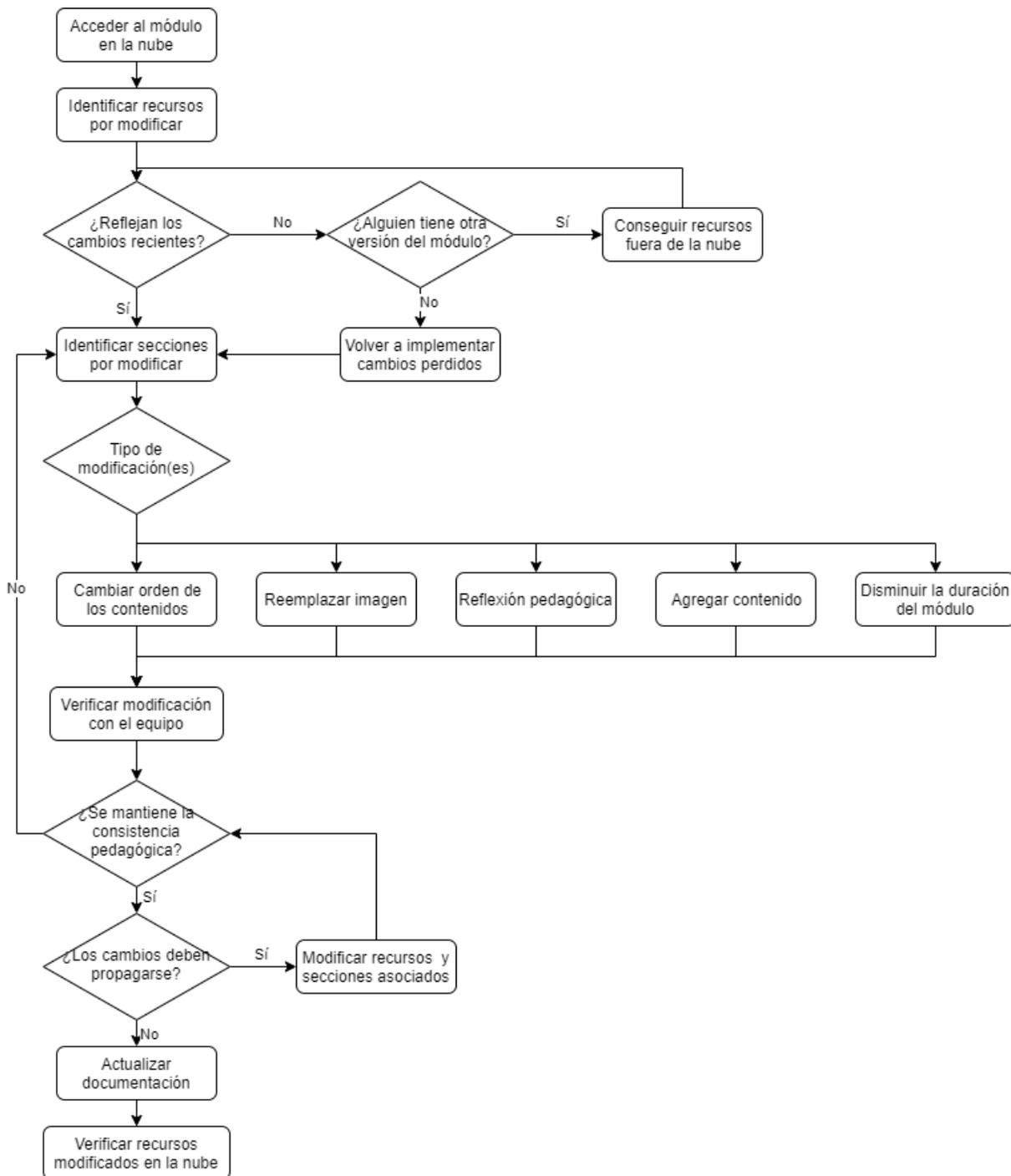


Figura 3.2: Flujo de modificación de módulos

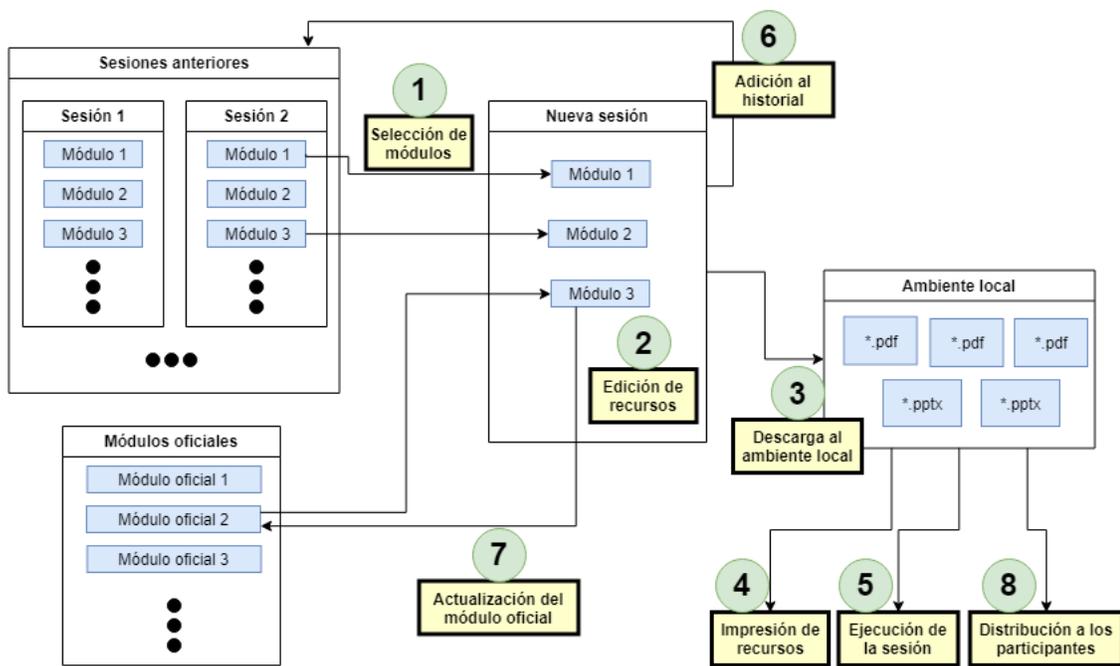


Figura 3.3: Flujo de creación de sesiones

Capítulo 4

Análisis de la situación actual

En este capítulo se estudia la situación actual a partir de las observaciones realizadas, buscando obtener una nueva perspectiva que permita identificar la causa raíz de los síntomas encontrados. Esto se realiza mediante modelos de los procesos o recursos y secuencias de preguntas relevantes que puedan aclarar el origen de las problemáticas, según la técnica de los cinco porqués, “The Five Whys Technique” en inglés [3], que busca causas raíces con cinco preguntas secuenciales. El proceso de las cinco preguntas no se presenta de forma explícita en este documento, sino que se presentan las preguntas clave, que entregan las conclusiones más relevantes.

El capítulo culmina con un replanteamiento de la situación actual y la enunciación de un desafío focal alineado con los objetivos, que aclare aspectos críticos de la solución por diseñar.

4.1. ¿Por qué se modifica?

Incluso en un escenario de estabilidad en los módulos, es decir, donde se ha probado que funcionan y el equipo sabe realizarlos, se identifican varias razones por las que el cuerpo de contenidos podría evolucionar, induciendo los problemas identificados.

Una sesión ofrecida por LeanSight en general dura media jornada hábil, es decir, alrededor de 4 horas. Los contenidos abordados durante esa jornada dependen de la necesidad que se identificó y conversó con el cliente. Según los expertos en el equipo, las habilidades que se busca entregar solo se adquieren después de profunda reflexión y vivencias propias, de manera que los contenidos deberían entregarse en un período de varias semanas.

Salvo las universidades, los clientes no planifican períodos tan largos para la formación de sus colaboradores y las ventanas de tiempo son extremadamente pequeñas. Así, uno de los desafíos del diseñador corresponde a ajustar los contenidos, que ya han sido probados con otras audiencias y fueron considerados estables, a la rendija de tiempo que el cliente proporciona, pero manteniendo el mismo nivel de impacto. Estas condiciones son la razón más frecuente de modificaciones en el cuerpo de contenidos y es tomada como contexto inalterable para nuestro análisis, pues viene de la mentalidad de los clientes y las presiones que tienen dentro de su organización, además de escaparse de los alcances de este trabajo.

El equipo de diseñadores educacionales se mantiene en constante mejora, formándose a través de cursos, libros y conferencias referentes a los tópicos que se enseñan en los talleres. El nuevo conocimiento, combinado con el existente genera reflexiones que ayudan a un mejor entendimiento del contenido de los talleres y la forma en que se pueden alcanzar sus objetivos de aprendizaje. Desde la reflexión se obtienen múltiples conceptos de mejora para los módulos, que deben ser adaptados según la audiencia y puestos a prueba en talleres presenciales. Cambios de este tipo hacen evolucionar el contenido frecuentemente, alterando la estabilidad de los recursos y contenidos. Muchas veces la reflexión ocurre sin conocimiento nuevo, y ayuda a adquirir conciencia sobre el cambio de paradigma que se persigue, mejorando la experiencia de aprendizaje y la efectividad de los talleres.

La parte visual de los talleres también mejora constantemente, definiendo una línea gráfica con cambios muy concretos y específicos, como colores o ilustraciones nuevas. Muchas veces se hacen prototipos rápidos, sin demasiado foco en su apariencia, dejando mucho espacio de mejora en este ámbito, y volviendo muy frecuente a este tipo de modificaciones.

Cuando se reflexiona sobre el perfil de los participantes y las tareas que el participante puede hacer o aprender, se utiliza el modelo de Zona de Desarrollo Próximo [6]. Este concepto surge de la teoría educacional de Vygotsky, que divide tareas según su dificultad para cierto aprendiz en tres zonas: lo que ya sabe hacer sin ayuda, lo que puede hacer con un guía o facilitador (desarrollo próximo), y lo que no puede hacer, incluso con ayuda. Es necesario conocer a los participantes para guiarlos siempre a través de su zona de desarrollo próximo y evitar entregar contenidos que no entiendan.

Finalmente, las modificaciones se pueden producir por un mal entendimiento del contenido actual, dañando la integridad del guion y dificultando el entendimiento del resto del equipo. Este origen de modificaciones debe ser evitado y ha llevado a LeanSight a varios intentos de sistematización del entendimiento colectivo, que serán detallados más adelante.

Con todo lo anterior, la labor de un diseñador educacional se sintetiza de la siguiente forma: “armar módulos educativos, a partir de contenido previo, para una ventana de tiempo dada, según un perfil específico de participantes y sin perder consistencia educativa”.

4.2. ¿Qué se modifica?

Aún queda hacer más concreta la lógica detrás de los talleres y proponer un modelo que lo describa de forma efectiva. Ya se ha expuesto que los recursos se modifican por varias razones, sería redundante decir que los recursos mismos cambian, pero tiene sentido preguntarse si es consecuencia de otros cambios más abstractos, o en qué resulta concretamente una mejor conciencia sobre el cambio de paradigma. Es necesario comprender el flujo completo de la información para entender el problema y brindar una solución efectiva.

Ya se introdujo el concepto de guion, que describe el flujo de un taller, pero no es siempre ese relato o los objetivos lo que cambia, sino un concepto de más bajo nivel y corresponde directamente a cada elemento del relato. En la dirección opuesta, hay también una razón por la que el relato tiene la forma que tiene, es lo que se busca lograr con el relato en general y las habilidades que se entregan, en otras palabras, los objetivos generales de aprendizaje,

competencias y principios. Así se concluye que es posible modelar los talleres de LeanSight en tres capas, como se muestra en la Figura 4.1.

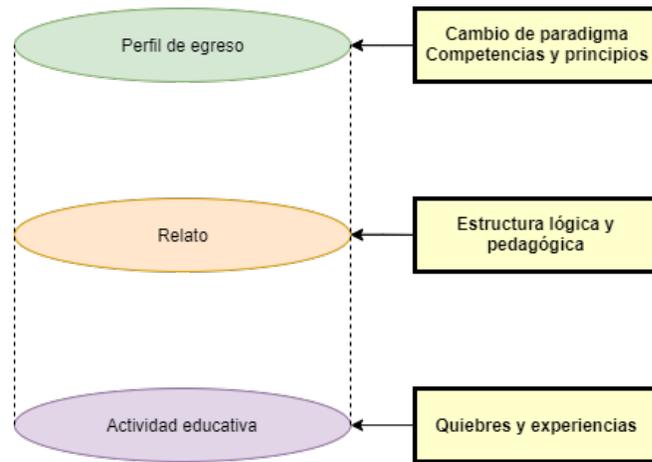


Figura 4.1: Modelo de tres capas para módulos educativos

La capa más alta, se construye en torno a un cambio de paradigma, describe sus principios y lo que se espera conseguir en los participantes de los talleres, en términos de competencias. Similar al *perfil de egreso* que definen universidades e institutos o al plan curricular que define el ministerio de educación, con objetivos específicos y actitudes.

La siguiente capa determina el *relato*, brindando una estructura lógica a cada taller. Aquí se aplican conocimientos de marketing, pedagogía y neurociencia para asegurar que el taller sea fácil de seguir y efectivo en su aplicación.

La capa más baja contiene cada sección de los recursos educativos, pero no se limita a ellos, corresponde a las *actividades educativas*. La capa es conceptual, por lo que describe y determina a las diapositivas y a las páginas que se usan en talleres, nunca al revés. Los elementos pertenecientes a esta capa son implícitos y difíciles de visualizar, pero se definen naturalmente al buscar la *intención* de cada actividad en los recursos educativos. Este concepto es nuevo para el equipo, pero fue comprendido y validado rápidamente al entender que su lógica siempre estuvo presente en los diseños previos.

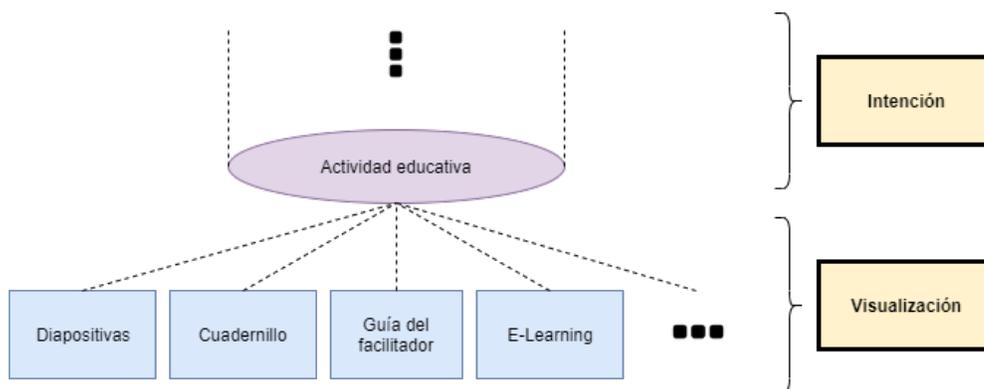


Figura 4.2: Distinción entre la intención y la visualización de una actividad

Cada parte de un recurso educativo es una *visualización* de cierta actividad educativa, que se define por su propia *intención*, como se muestra en la Figura 4.2. La intención específica es lo que se quiere lograr en el participante como parte de un proceso cognitivo. La actividad educativa se materializa a través del uso de un instrumento, por ejemplo, diapositivas o videos, un cuadernillo en el que el alumno puede escribir, o incluso una actividad interactiva a realizar por éste, si se trata de e-learning. En otras palabras, todo lo que experimenta un participante es parte de una actividad específica, cada una con su propia intención educativa.

Este diseño fue puesto a prueba con un prototipo que se detalla en el capítulo de validación (Sección 7.2)

4.3. Recursos de documentación

Los síntomas identificados son tan frecuentes que ya se han implementado varias formas de prevenirlos, solucionarlos o simplificarlos. La principal manera corresponde a la documentación de la estructura de los módulos, describiendo de alguna manera los objetivos de aprendizaje y la implementación de actividades en cada recurso. La documentación de módulos ha pasado por muchas formas y etapas, en particular para los recientes proyectos de e-learning ha tenido una forma propia, por tener clientes diferentes y tipos de recursos con los que no se tenía experiencia previa.

La documentación no ha tenido un objetivo claro, pues se basa en la intuición de que se requiere alineamiento en el equipo y entendimiento sobre los contenidos en todo momento. Sin un objetivo, es difícil declarar si cada experimento es o no un fracaso, pero sin duda se pueden rescatar aprendizajes de cada uno.

Distintas terminologías aparecen entre estos recursos, producto de la adquisición de nuevo conocimiento sobre pedagogía. En general se definen “objetivos” o “metas”, principios a entregar, paradigma por cambiar, quiebres a realizar y otros detalles. El recurso de documentación que más esfuerzo recibió por parte del equipo y de principal interés para este trabajo, es el reciente Modelado de Módulos Educativos (MoME). Sin embargo, ese recurso quedó rápidamente obsoleto cuando el equipo se enfocó en el desarrollo de soluciones e-learning, que implicó una rápida evolución del contenido.

El MoME es fruto del trabajo de título del ingeniero Lucas Puebla. Éste consiste en una tabla donde cada columna es un recurso educativo, y cada fila es una sección muy pequeña del guion, tan pequeña como sea posible. De a poco se fueron agregando objetivos de aprendizaje y cambios de paradigma, todo en el mismo lugar. Esta implementación tuvo buenos resultados, pero después de un tiempo quedó en desuso. Esto ocurrió de manera similar a sus predecesores, y vale la pena estudiar la razón, al igual que los aprendizajes que nos entrega.

El MoME es consistente con la forma en que se hacen planes curriculares tradicionalmente; es decir, una lista de contenidos (filas), con varios tipos de actividades (columnas) como, por ejemplo, clases, laboratorios, tareas, exámenes. La forma de este documento se mantiene dentro de la estructura de capas propuesta en la sección anterior, donde cada fila sería una actividad educativa y sus objetivos corresponden a las intenciones.

Durante el desarrollo del paradigma de e-learning nacieron nuevos recursos de documentación de forma espontánea, dado que era necesario comunicarse con nuevas personas; por un lado, los clientes que mantienen foco en los objetivos de aprendizaje e inicialmente, poco interés sobre la implementación de los cursos, y por otro lado los distintos colaboradores especializados, como el diseñador instruccional de SCORM y el locutor que graba las narraciones. Uno de estos recursos de documentación es la Ficha de Módulo, que servía para comunicarse con el cliente y darle una idea general de lo que sería el curso, cuánto duraría y qué contenidos se cubrirán. Otro recurso es la Plantilla de Implantación, que detalla de forma rigurosa la implementación de cada texto, imagen y actividad interactiva. Estos recursos resultaron ser de gran utilidad práctica para el equipo, pero reemplazaron momentáneamente al MoME, y por lo tanto este último dejó de ser actualizado por un tiempo. Un par de semanas bastaron para que el MoME estuviera obsoleto en documentación y los nuevos recursos tuvieran también inconsistencias.

Se observa que una de las principales causas de inconsistencias viene de una incorrecta o inexistente actualización del recurso de documentación. Estudiando sus causas, se observa que esta situación se debe al retrabajo que implica el mantener estas actualizaciones al día, porque cada cambio que se deba realizar sobre el material debe también documentarse en el MoME. El proceso es tan tedioso y propenso a errores humanos, que las inconsistencias aparecen más rápido de lo que se pueden solucionar.

4.4. Reformulación del escenario de trabajo

Con esta nueva perspectiva, los problemas encontrados se pueden articular de otra manera, asociándose al modelo en tres capas la necesidad de documentar el conocimiento y la situación actual de LeanSight. Esto corresponde a la causa raíz de los síntomas expuestos, y enmarca los caminos a posibles soluciones.

El Perfil de Egreso está completamente implícito en el equipo; aunque el equipo se encuentra alineado en torno a la entrega de competencias nucleares, el objetivo de alto nivel no está articulado en ningún recurso.

En recursos de documentación pasados, se registraba la función de cada sección dentro del relato desde el modelo ADKAR de marketing y la teoría educacional de Anderson y otros sobre la taxonomía de Bloom [1], que clasifican habilidades cognitivas en siete niveles de conocimiento [7]. Se sabe que existe un relato subyacente al relato, pero es costoso mantenerlo explícito y actualizado; por ahora no se encuentra documentado de forma concreta ni confiable en ningún recurso.

Ya que el diseñador educacional requiere adaptar talleres con frecuencia, se ve constantemente enfrentado al desafío de mantener la esencia del módulo en una ventana reducida. Estos recortes generan recursos duplicados e inconsistentes con gran rapidez, son difíciles de encontrar y entender, y son una fuente importante de retrabajo. Lo que modifica el diseñador conceptualmente es el relato o las actividades, pero actualmente realiza los cambios en el recurso mismo y sin documentarlo.

Cuando se realiza una modificación en un recurso, típicamente fue la intención la que

originalmente cambió. Hacer esto sin documentarlo induce a error y hacerlo de otra manera hace que el resultado esté completamente desconectado de los objetivos de aprendizaje. Aquí es donde se encuentran los problemas de edición y propagación. En concreto las cuatro acciones Crear, Leer, Actualizar y Eliminar elementos (CRUD en inglés) son tediosas e inducen a error por iniciarse sobre el elemento indebido.

Vimos que cuando los recursos cambian, probablemente ocurrió un cambio en otra capa, y por lo tanto se debe entender en cuál fue para mantener la consistencia y el foco. En la situación actual no se hace esta distinción, y el proceso que se sigue empieza con la edición de un recurso, sigue con la revisión de su intención, actualización de su documentación y finalmente, su propagación al resto de los recursos. El recorrido sube en abstracción y luego vuelve a bajar, en contra de la lógica tácita y sin considerar que pudo haber un cambio mayor en los objetivos de aprendizaje, el cual no fue correctamente implementado.

La lógica propuesta en torno a actividades en vez de archivos replantea la forma en que se crean sesiones, pues cubre la causa raíz de sus problemáticas. Las modificaciones que se hacen para distintas realizaciones de los talleres existentes son mejoras que se deberían mantener, o bien, eliminaciones de actividades para hacer calzar el taller a la duración acordada. Mantener estos cambios en distintos archivos conlleva varios de los síntomas que el administrador de sesiones cubría. Entonces, la implementación de sesiones requiere ahora una investigación adicional, y de seguro será considerablemente más sencilla que en el modelo anterior.

4.5. Desafío focal

La situación actual se caracteriza por tres problemas clave, que constituyen la causa raíz de los síntomas identificados: (1) el perfil de egreso no está bien definido, (2) el relato subyacente sigue implícito y su documentación desactualizada y (3) la modificación en los recursos está desconectada de su lógica e intención.

Los síntomas concretos que estos problemas incluyen a la mayoría de los ya expuestos: tiempos de edición de hasta 2 horas para cambios conceptualmente sencillos, decenas de minutos en encontrar versiones específicas de recursos, una cantidad inmanejable de recursos duplicados, importantes mejoras olvidadas entre las copias de los módulos, confusiones durante los talleres, prototipado lento, horas de trabajo en hacer recortes que ya se han hecho anteriormente, y extenso retrabajo por errores de propagación de cambios.

Las problemáticas están distribuidas en las tres capas, y abordarlas todas se escaparía del alcance de este trabajo. Sin embargo, la prioridad se vuelve clara cuando se considera que las prácticas de la situación actual han inhabilitado cualquier solución en alguna de las dos capas más altas. Por lo tanto, es necesario primero conectar los recursos con la lógica subyacente para habilitar la implementación de mejoras sostenibles en niveles de abstracción más altos.

Encontramos entonces el desafío focal de este trabajo, que persigue crear una conexión sostenible entre las actividades educativas, sus intenciones y sus visualizaciones, simplificando el proceso de edición y habilitando una visión de alto nivel sobre el cuerpo de contenidos.

Capítulo 5

Concepción de la solución

En este capítulo se describen las principales decisiones de diseño para la posterior implementación de la aplicación. Se incluye una visión general de la aplicación, su interacción con los principales actores, el diseño lógico de la plataforma, fundamento sobre la decisión de herramientas de software para la generación de documentos y el framework de la aplicación, y finalmente el modelo de datos consecuente con el planteamiento.

5.1. Plataforma de edición

Según la arquitectura lógica propuesta, cuando se quiere modificar un recurso, deberíamos considerar la naturaleza de la modificación y la capa en que se ubica, comprendiendo que todos los niveles de abstracción inferiores se verán afectados. Por ejemplo, un replanteamiento en el cambio de paradigma que se persigue con el módulo modificaría el relato completo y probablemente requeriría nuevos elementos también. Por esto, en el modelado de los contenidos deberían primar las actividades como unidad indivisible.

Una de las restricciones más relevantes de esta problemática es que la conexión entre las actividades y los recursos debe ser robusta. No es admisible que tal conexión quede inconsistente o desactualizada al final de un proceso de edición, porque se volvería a los problemas que ya se conocen, invalidando la solución por padecer de los mismos defectos identificados en el MoME.

Esto solo deja dos alternativas, se habilita la actualización automática de la documentación de las actividades y sus conexiones con los recursos, o bien, se habilita la generación de los recursos a partir de las actividades.

La naturaleza de los archivos que conforman el cuerpo de contenido actualmente es inherentemente contraria a la noción de actividad que hemos descubierto, porque una misma actividad es encontrada a través de todos los archivos de forma transversal, y un archivo solamente toma su forma final cuando todas las actividades que lo conforman han sido determinadas. En otras palabras, las filas del plan curricular están desalineadas entre sí. Varios de los síntomas identificados provienen precisamente de esta estructura, pues es de difícil navegación y comprensión.

De lo anterior se concluye que, si se sigue utilizando Google Docs y Google Slides o los productos de Microsoft Office como el punto de partida de la conexión que se persigue, la conexión con las actividades es poco natural; a esto lo llamaríamos “foco en las columnas” o “foco en las actividades” de acuerdo con la estructura de los planes curriculares. Las inconsistencias serían difíciles de detectar y se tendría poco control durante la propagación de cambios y la navegación y edición del contenido. Entonces el siguiente paso consiste en habilitar la generación de documentos a partir de las actividades, es decir, con foco en las filas del plan curricular. Los recursos podrían seguir siendo del mismo formato, pero el primer nivel de navegación se debe encontrar en las actividades, así que se requerirán herramientas de automatización para lograrlo. De cualquier modo, las alternativas deben ser evaluadas primero.

5.2. Generación de documentos

Independientemente del formato que se utilice, es crítico que la generación de documentos sea fácil de usar. Por ejemplo, dividir un documento en tantos archivos como actividades tenga, no es una solución viable si se deben editar los archivos mismos, porque sería altamente costoso encontrar, navegar y modificar cada actividad. No se puede perder de vista el objetivo de minimizar el retrabajo, y en general, el tiempo usado en modificar y crear módulos.

Se vuelve necesaria una plataforma que mantenga todo en un mismo lugar, adecuadamente seccionado por actividades y permitiendo la adición de futuras funcionalidades que faciliten el trabajo sobre el cuerpo de contenidos. Es importante notar que esto no impide el uso de los formatos actuales como resultados, pero sí impide que sean el origen de las visualizaciones.

Esta idea de solución se simplifica al introducir el concepto de sintaxis concreta y abstracta, en conjunto con un parser y un intérprete, determinando un conjunto de decisiones por tomar. La situación actual incluye todas las partes en la misma herramienta, otras implementaciones, en cambio, hacen la distinción, como por ejemplo LaTeX o Markdown.

Las alternativas para procesar la sintaxis (parser e intérprete) determinan en parte la sintaxis misma, por lo que la modularidad no podrá ser tan estricta, las opciones son diversas y se discutirán a continuación. Microsoft Office cuenta con interfaces para crear y manipular el contenido de sus documentos en JavaScript, Visual C # y Visual Basic. Google ofrece Google Apps Scripts, un producto que brinda un ambiente de automatización sobre sus productos basado en JavaScript, incluyendo el manejo del almacenamiento basado en nube, Google Drive, y también ofrece una API REST, que es considerablemente más rápida al procesar solicitudes, pero entrega una interfaz compleja para manejar el contenido.

Dada una sintaxis específica, se puede también implementar un desarrollo a medida que entregue resultados que cubran las necesidades de un facilitador, o sea, un cuadernillo y un apoyo visual o algún equivalente. Pandoc¹ es una herramienta de software gratuita que permite conversiones entre numerosos lenguajes de notación, y requeriría un trabajo posterior para darle a los documentos un estilo adecuado. Existen también alternativas para MS Word y PowerPoint, las cuales están pensadas específicamente para desarrolladores, siendo GitPitch²

¹<https://pandoc.org/>

²<https://gitpitch.com/>

un ejemplo representativo, entregando una simple integración con control de versiones y una sintaxis sencilla basada en Markdown para crear diapositivas. Finalmente, existen editores de LaTeX ya desarrollados que cuentan con las funcionalidades necesarias para implementar diapositivas y documentos.

Ya que los usuarios de la solución a desarrollar no serán programadores, si no se les entrega un editor visual, necesitarán una sintaxis muy sencilla para describir el cuerpo de contenidos. LaTeX, Markdown, HTML e incluso XML ofrecen sintaxis específicas para crear documentos, que pueden posteriormente ser tratadas a medida con otras herramientas, como las ya mencionadas. LaTeX ofrece flexibilidad y apunta a un usuario familiarizado con la programación o habilidades similares. Por su lado, Markdown sacrifica flexibilidad para perseguir una fácil lectura y escritura a primera vista. HTML y XML no son accesibles para los usuarios a los que se apunta, por ser muy aparatosas, difíciles de leer y propensas a errores. Incluso con lo anterior, es siempre una alternativa válida implementar una sintaxis a medida, que puede adaptarse muy bien al usuario final, aunque tomaría mucho tiempo implementarla.

Al decidir la sintaxis junto al equipo, se le dio prioridad a Markdown por sobre LaTeX, por ser mucho más simple y legible, aunque no cuente con las mismas capacidades de automatización de contenido. Se identifica también que el desarrollo de una sintaxis a medida entregaría un nivel de solución similar al de Markdown, pero con un tiempo de desarrollo considerablemente mayor, de cualquier manera, se puede adaptar la sintaxis para las necesidades que no logre cubrir, sin necesidad de crear una desde cero. Con esto, se elige Markdown como base para la sintaxis a utilizar.

La solución tiene como requisitos críticos (1) que el contenido se pueda manejar de forma automatizada, para facilitar propagación de cambios, y (2) la posibilidad de integrar varios documentos de forma transversal, es decir, en torno a las actividades asociadas y no a los archivos por separado. De esto eliminamos directamente las alternativas como GitPitch y similares, pues manejan un único tipo de documento a la vez.

Otros requisitos relevantes pueden ser cumplidos parcialmente, pero siguen siendo importantes; por ejemplo, el servicio debe ser confiable, es decir, debe estar disponible la mayor parte del tiempo (no caerse) y en distintos soportes. Se necesita soportar imágenes de varios tipos, pues se sabe dentro del equipo que las imágenes cumplen funciones pedagógicas cruciales. El manejo de los archivos y de la sintaxis deben ser lo suficientemente simples para que los usuarios puedan comprenderlo en poco tiempo. Algunas alternativas requerirán más tiempo de desarrollo que otras, y esto es considerado como parte de las dimensiones a evaluar.

Para comparar las opciones de parser e intérprete, se le asignó puntaje a cada variable relevante para la solución y el porcentaje de cumplimiento de cada alternativa, como se muestra en la Tabla 5.1. Los puntajes nacen producto de la investigación de la primera sección y han sido consultados con el equipo de manera cualitativa antes de proceder. Se incluyó la edición en LaTeX, por ser un editor y conversor válido que podría utilizarse como interfaz, o se podría pagar el costo para capacitar en el uso de LaTeX si resulta ser una mejor solución. El nivel de solución considera características como la usabilidad, el tiempo que tomaría realizar tareas específicas en su uso y la relación que tiene con las prácticas actuales de LeanSight, es decir, la capacidad de resolver las problemáticas identificadas.

	Puntaje	Google	Edición en LaTeX	Microsoft Office	Editor basado en Markdown y Pandoc
Nivel de solución	30	30 %	50 %	30 %	80 %
Confiabilidad	30	20 %	100 %	60 %	80 %
Soporte de gráficas	15	80 %	70 %	100 %	70 %
Curva de aprendizaje	15	100 %	0 %	80 %	80 %
Tiempo de desarrollo	10	20 %	80 %	0 %	0 %
Puntaje total	100	44	63,5	54	70,5

Tabla 5.1: Evaluación de alternativas de editor de documentos

La evaluación mostrada en la Tabla 5.1 considera que los editores de Google y su servicio de nube han fallado frecuentemente en momentos críticos; que seguiría tomando mucho tiempo realizar ciertas tareas en los editores actuales (Google y Microsoft); que LaTeX es considerablemente difícil de aprender para el usuario final; que seguir usando editores de Google, sin Markdown sería simple de usar, pero no soluciona efectivamente las problemáticas; y que un editor de LaTeX requeriría poco tiempo de desarrollo para que empiece a funcionar.

Los editores de Google con una sintaxis personalizada no fueron incluidos, porque pruebas anteriores han demostrado que en esa situación los usuarios siguen editando los archivos de salida, en vez del texto fuente, haciendo persistir las problemáticas.

Con esta evaluación, se decide que la mejor alternativa es un editor desarrollado a medida, pero evitando implementar una sintaxis nueva. Pandoc ofrece varios formatos de salida adecuados, como revealjs, que se basa en una librería de JavaScript para genera presentaciones de diapositivas en HTML, y el motor wkhtmltopdf, que a través de HTML convierte texto escrito en Markdown en un documento PDF.

5.3. Estructura conceptual de la solución

Con lo anterior, se pueden identificar componentes críticas para la solución. Un diseñador educacional debe ser capaz de administrar módulos e imágenes, navegar y editar el contenido según actividades y previsualizar el resultado de sus cambios. Cada funcionalidad cuenta con fundamento desde la investigación realizada y se expone en la sección 6.2 en detalle.

La Figura 5.1 muestra la relación entre los distintos tipos de usuarios de la aplicación y su interacción con las distintas componentes de software. El principal usuario de la aplicación es el Diseñador educacional, responsable del diseño e implementación de contenidos. Se ofrece una biblioteca de imágenes y una de módulos, para entregarle al diseñador la capacidad de crear, consultar, actualizar y eliminar módulos o imágenes. El contenido debe ser editable de acuerdo con el diseño planteado y la edición y la navegación deben ocurrir en torno a las actividades; se ofrece un editor de actividades para cada actividad y una lista de actividades que habilite su fácil navegación, esto es particularmente útil para el experto, que debe asegurar el alineamiento de las intenciones con los objetivos de aprendizaje. Desde el editor debe ser

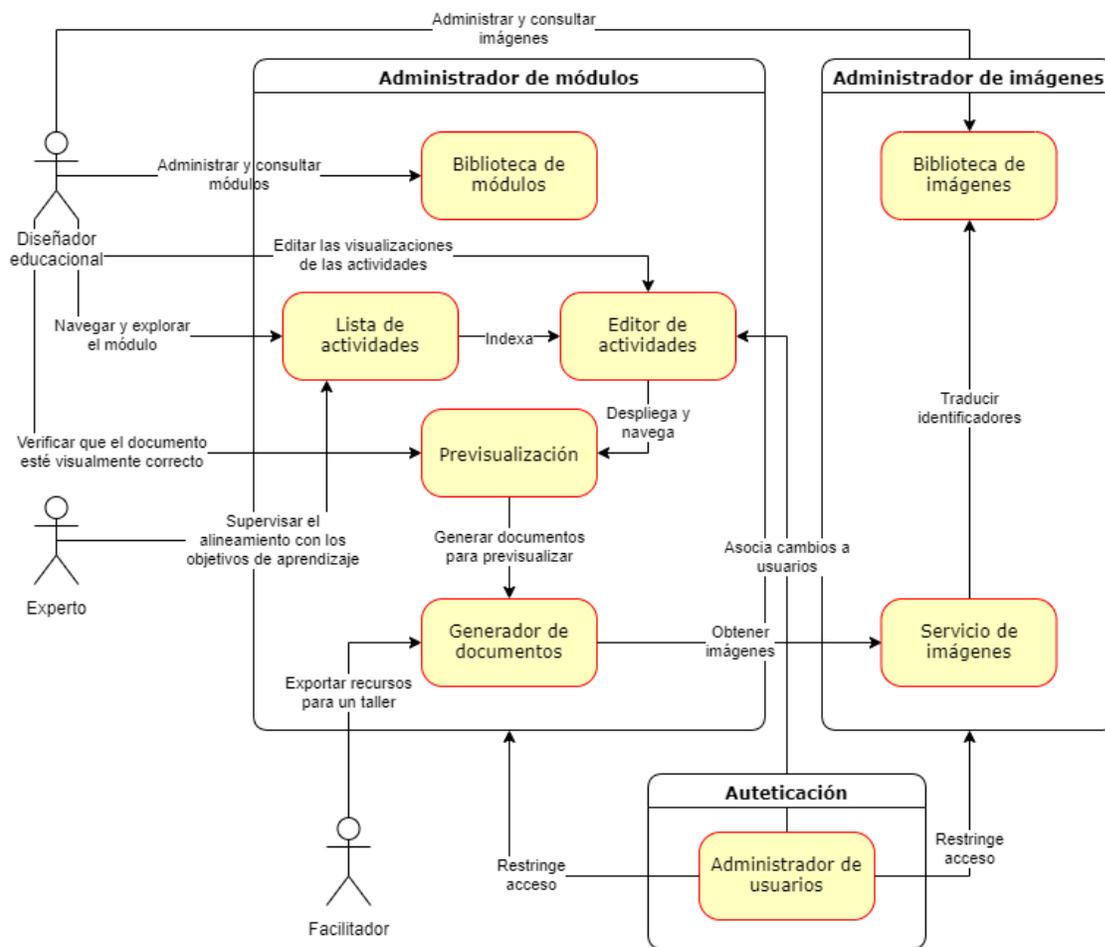


Figura 5.1: Diagrama de interacciones entre actores y componentes de la aplicación

fácil y rápido visualizar el efecto de las modificaciones sobre los archivos finales en algún tipo de previsualización.

La aplicación debe restringir acceso a cualquier persona que no esté adecuadamente autenticada y cada cambio debe ser asociado a quien lo realizó. Se diseñará un sistema de administración de usuarios simple que implemente estas funcionalidades

Por último, el facilitador debe poder obtener los documentos que necesita con poco esfuerzo y sin tener que conseguir módulos actualizados. El generador de documentos se encargará de crear la previsualización y los recursos exportados que se utilizarán en el taller, apoyándose en el servicio de imágenes para obtenerlas a partir de una sintaxis legible y cómoda.

5.4. Framework para la aplicación

Las herramientas de software escogidas deben ser integradas junto a la lógica propuesta en una interfaz para los usuarios, y un modelo de datos para almacenar el cuerpo de contenido. Para implementar ambos existen numerosas alternativas y el criterio para escoger será encontrar la que tenga la mejor curva de aprendizaje, legibilidad de código e integración con la base de datos.

Teniendo cierta experiencia previa con el framework de desarrollo web Django³, se reconoce que cuenta con las funcionalidades necesarias, habilitando un desarrollo rápido, un diseño práctico, alta extensibilidad y permitiendo que nuevos programadores se puedan incorporar al desarrollo con pocos obstáculos. El equipo está de acuerdo con la elección de framework, los programadores del equipo lo conocen, y por lo tanto se habilitará el entendimiento compartido del código, evitando la dependencia absoluta del autor de la aplicación para su mantención.

5.5. Modelo de datos

Como se explicó en el replanteamiento de la situación actual, las sesiones tienen que ser reinterpretadas, con una nueva investigación, posterior a la implantación de esta solución; sin embargo, su modelo se incorporará según su concepción inicial, buscando que cuente con suficiente extensibilidad como para poder agregar las funcionalidades necesarias en el futuro.

De esta forma, y ya con el problema acotado a la capa más baja del modelo anteriormente planteado, el modelo se centra en los módulos con sus respectivas actividades, cada una con su propia intención y la implementación de sus visualizaciones, que convenientemente se decidió que estuvieran escritas en Markdown, o sea, basadas en texto.

Se discutió que las actividades cumplen un propósito que podría ser utilizado para introducir o reforzar varios conceptos diferentes, es decir, una actividad puede estar asociada a varios módulos a la vez. Entonces los módulos y las actividades tienen una relación de muchos a muchos en el modelo.

La tabla intermedia para la relación entre módulos y actividades contiene información muy importante para la integridad del contenido. Esta información específica, por ejemplo, el orden de las actividades y si éstas se encuentran ocultas o no, siendo esta última manejable por el lado del cliente, pero se prefiere almacenar pensando en que puede ser justamente la representación de sesión que se necesite en el futuro. La importancia de esta tabla y las posibilidades que ofrece, se evidencian con mayor facilidad en el trabajo a futuro (sección 8.2).

La aplicación busca también modularizar el uso de imágenes, por lo que requiere su propio modelo y, por conveniencia, su propia aplicación en Django. El modelo debe contar con una clave que identifique a cada imagen, además de dos versiones de la misma imagen: (1) la imagen original y (2) una versión pequeña para optimizar la carga de previsualizaciones. La versión liviana de la imagen podría ser generada en el momento, pero es preferible que se almacene, como usualmente se sugiere para aplicaciones web, se debe considerar el procesamiento costoso y el almacenamiento barato.

Las imágenes, los módulos y las actividades deben implementar las cuatro operaciones CRUD para cubrir por completo la problemática, entendiendo que acciones irreversibles requerirán una confirmación por parte del usuario. Como trabajo a futuro, es relevante, mas no crítico, mantener trazabilidad de estas acciones, pudiendo rápidamente consultar el responsable de las últimas modificaciones al cuerpo de contenidos. Lo anterior da lugar al modelo de datos ilustrado en la Figura 5.2 que será explicado en detalle a continuación.

³<https://www.djangoproject.com/>

Módulos (Module) y actividades (Activities) tienen entre sí una relación de muchos a muchos, porque los módulos se componen de actividades y las actividades son en principio reutilizables. Como se necesita saber para cada actividad dentro de un módulo su ubicación y si está oculta o no (sección 6.2.12), se requiere una tabla intermedia en esta relación, llamada ModuleActivity, que representa una actividad dentro de un módulo específico y en una posición específica.

Las actividades usan imágenes (ImageResource) en su implementación, es decir, contienen identificadores (key) dentro de sus campos slides y booklet. Como los identificadores están dentro del texto, la funcionalidad que administra imágenes (ilustrada en la sección 5.3) se debe encargar de traducir los identificadores a una dirección que permita su uso.

El modelo de usuarios (User) es principalmente útil para restringir acceso a la plataforma, pero se aprovecha también su existencia para almacenar al realizador del cambio más reciente sobre cada actividad o al autor de un comentario (Comment) sobre una actividad en un módulo. El usuario corresponderá a la implementación de Django para usuarios.

Las sesiones (Session) corresponden a un conjunto ordenado de módulos, realizados en una fecha específica para un cliente (Client) en particular, que debe estar a su vez adecuadamente identificado como contraparte.

Con la investigación realizada se entiende que este modelo puede crecer más allá de lo que se enuncia o su implementación puede ser priorizada según lo que más valor aporte a los usuarios. Cada posible cambio en su forma o prioridad se encuentra adecuadamente documentado en la sección de trabajo a futuro (sección 8.2).

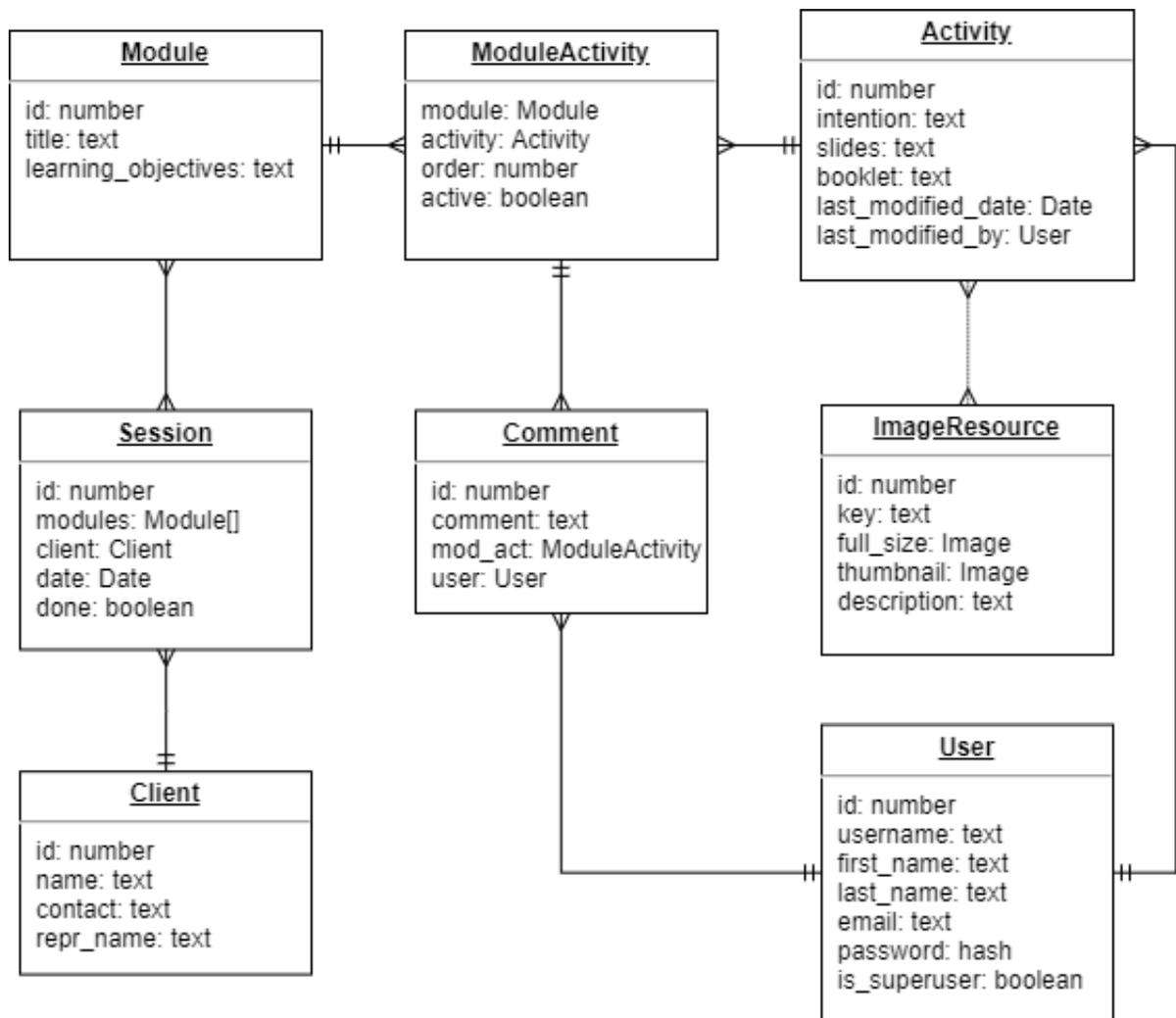


Figura 5.2: Modelo de datos de la solución

Capítulo 6

Implementación de la solución

Con el desafío focal y el modelo definido, se pueden decidir funcionalidades que actúen como contramedidas para las problemáticas identificadas. La implementación no se trabajó como un solo gran proyecto entregado luego de su completa ejecución, sino que se fue poniendo a prueba cada funcionalidad adicional a partir de un punto estable de la aplicación, y exponiendo al equipo a avances semanales, pretendiendo así detectar oportunamente pérdidas de foco o desalineamiento con la problemática.

6.1. Navegación en la aplicación

La Figura 6.1 muestra la forma en que las distintas vistas y funcionalidades se relacionan entre sí. Principalmente se compone de una vista de inicio de sesión (login), vistas de bibliotecas de módulos e imágenes y el editor de actividades basado en Markdown. Otras funcionalidades incluyen previsualización de actividades y exportado de módulos como recursos utilizables en un taller. El detalle de cada funcionalidad se describe a continuación.

6.2. Funcionalidades

La implementación se expone a continuación a través de las distintas funcionalidades de la aplicación web sin un orden específico, brindando contexto suficiente para comprender el problema que intenta abordar y las creencias que busca confirmar.

6.2.1. Editor

El uso de Markdown no solo simplifica la escritura, la legibilidad y el almacenamiento de las visualizaciones, sino que también elimina la complejidad propia de editores de texto de tipo WYSIWYG¹ (sigla en inglés para “lo que ves es lo que obtendrás”), ya que el estilo del documento se describe a través del texto mismo.

El editor consiste simplemente en campos de texto separados para cada visualización.

¹<https://whatis.techtarget.com/definition/WYSIWYG-what-you-see-is-what-you-get>

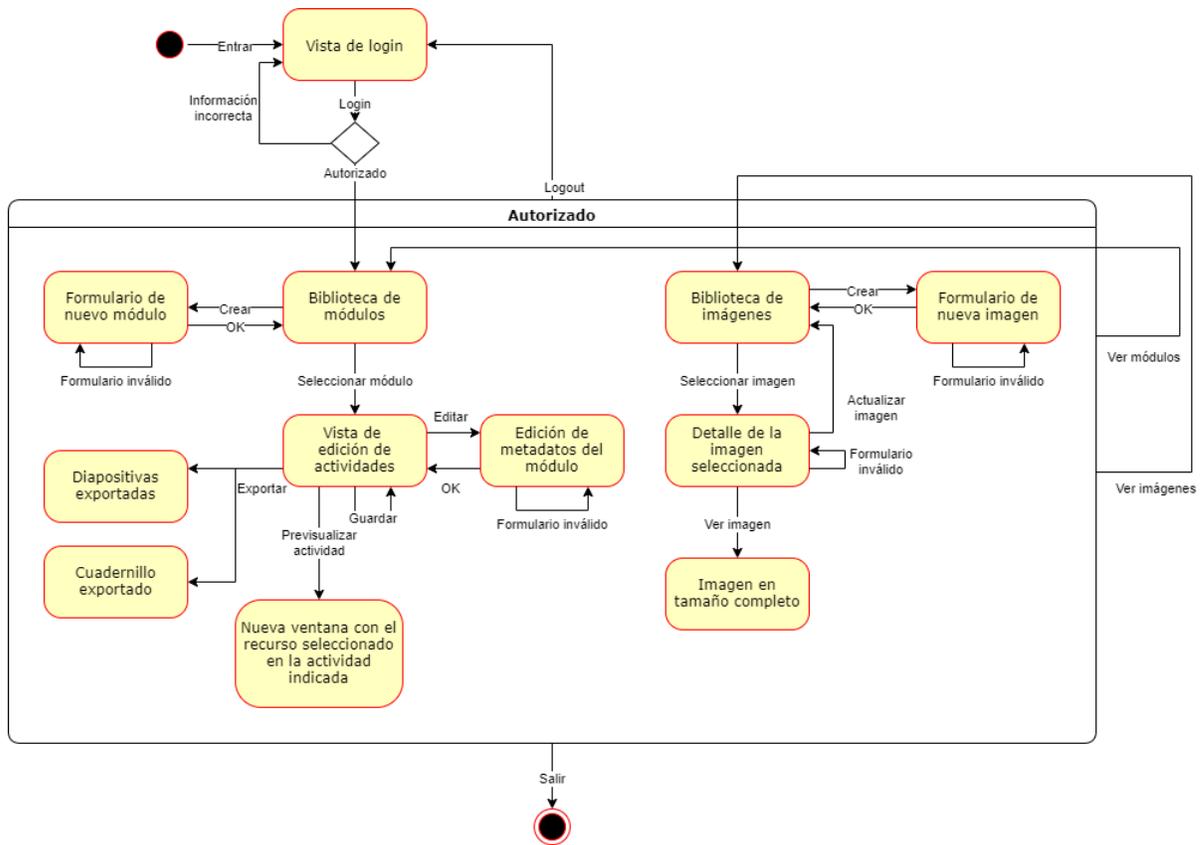


Figura 6.1: Mapa de navegación de la aplicación

El procesamiento del texto y la generación de los archivos correspondientes se encuentra modularizada y fuera de esta interfaz.

El objetivo del editor es que el diseñador educacional pueda mantener el foco en el desarrollo de contenidos, sin que el diseño gráfico entorpezca su trabajo. Esto permite realizar prototipos rápidos y flexibilizar el flujo de trabajo sin perder alineamiento con la intención de cada actividad.

La Figura 6.2 muestra el editor basado en Markdown para una actividad específica y con su sintaxis para insertar imágenes, títulos y líneas horizontales para que los alumnos escriban. Más adelante en este capítulo se muestra el resultado de esta implementación. Cada actividad cuenta con su propio editor, para conseguir el “enfoco en las filas” que se enuncia en el desafío focal.

6.2.2. Generador de documentos

Pandoc es el motor encargado de convertir el contenido escrito en Markdown, a un formato que se pueda utilizar en un taller o formación. Este motor está implementado en Haskell y admite numerosos formatos de entrada y salida, junto a la posibilidad de elegir motores para crear documentos PDF y una conveniente librería de Python.

Las diapositivas se convierten a revealjs, un formato escrito en HTML que en conjunto

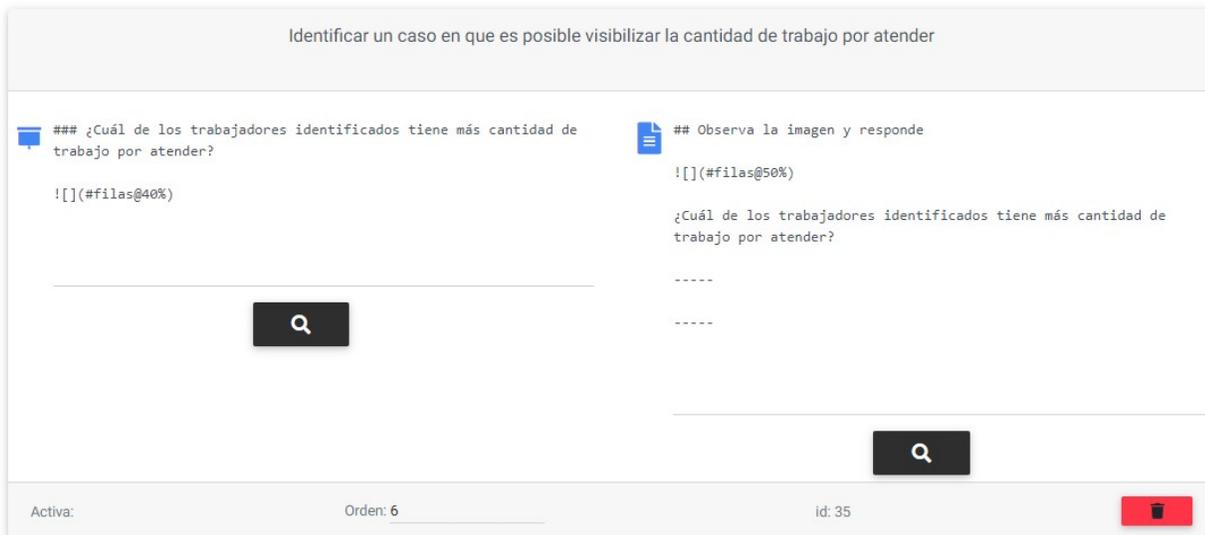


Figura 6.2: Editor basado en Markdown

con una librería de JavaScript entregan presentaciones de diapositivas. El formato es simple en su estructura, incluso en HTML y la traducción desde Markdown resulta muy natural, con cada título como inicio de una nueva diapositiva y manteniendo la sintaxis de Markdown para links e imágenes.

Se tienen varias opciones de motores para crear los cuadernillos en formato PDF, pero por conveniencia se utilizará wkhtmltopdf, ya que permite crearlos a partir de un HTML, sobre el que se puede tener completo control mediante un archivo CSS.

Pandoc permite aplicar filtros sobre su AST², habilitando implementaciones personalizadas dentro del lenguaje de origen o hacia el formato de salida. Se concluye que es una herramienta de software lo suficientemente poderosa y robusta para la tarea que se persigue.

Las Figuras 6.3 y 6.4 muestra el resultado de la implementación presentada en la Figura 6.2, se puede observar la modularidad del manejo de imágenes, la simplicidad de la implementación en Markdown y cómo habilita el foco en el contenido sin tener que tomar muchas decisiones de diseño gráfico. Más adelante en esta sección se explica el diseño de la inserción de imágenes.

6.2.3. Lista de actividades

Parte de los síntomas identificados es que toma mucho tiempo encontrar el lugar en que se debe realizar cierta modificación específica; incluso teniéndola en frente, es necesario preguntarse si la actividad tiene la intención que se cree antes de realizar cualquier cambio.

La aplicación ofrece una interfaz de usuario simple, donde los servicios se acceden a través de una lista con las intenciones. Además, el usuario se puede desplazar libremente a lo largo de un módulo, observando las implementaciones de cada actividad, mientras que en el menú de navegación se destaca la actividad que se está observando actualmente.

²<https://pandoc.org/filters.html>

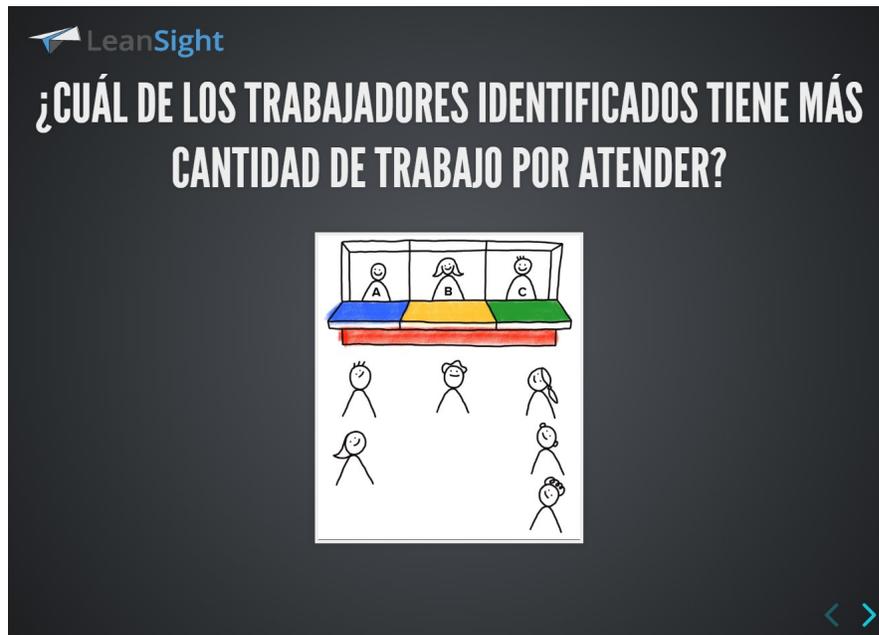


Figura 6.3: Ejemplo de diapositiva generada desde Markdown

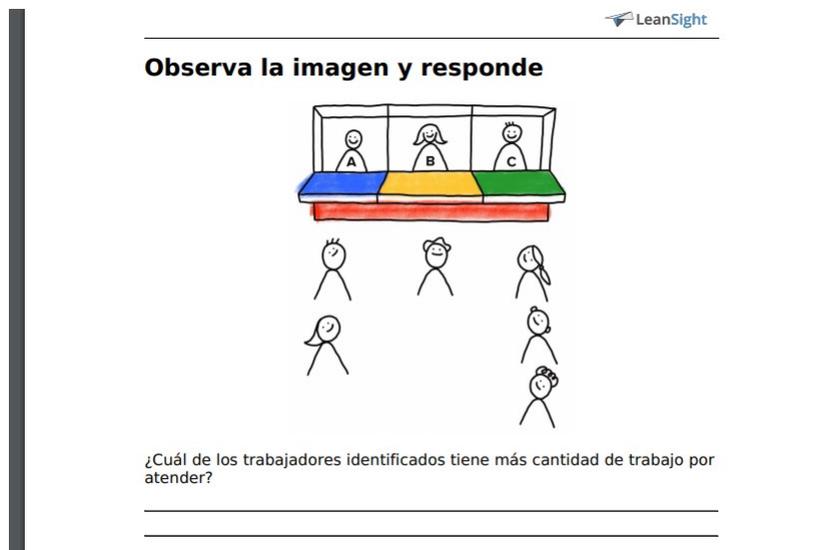


Figura 6.4: Ejemplo de cuadernillo generado desde Markdown

La interfaz de navegación del contenido, en conjunto con el editor, permite seguir en paralelo las distintas implementaciones de cada actividad, sin tener que navegar archivos separados para modelar o modificar un contenido con una intención específica. Así se pretende disminuir sustancialmente el tiempo de navegación y, por consiguiente, el tiempo de edición en general.

La Figura 6.5 muestra a la izquierda la lista de actividades y a la derecha las actividades mismas. La lista muestra la intención de cada actividad, siendo cada una clickeable para desplazar la vista de la derecha hasta la ubicación correspondiente a tal actividad. La actividad que se está visualizando actualmente se encuentra destacada en la lista, para que el usuario

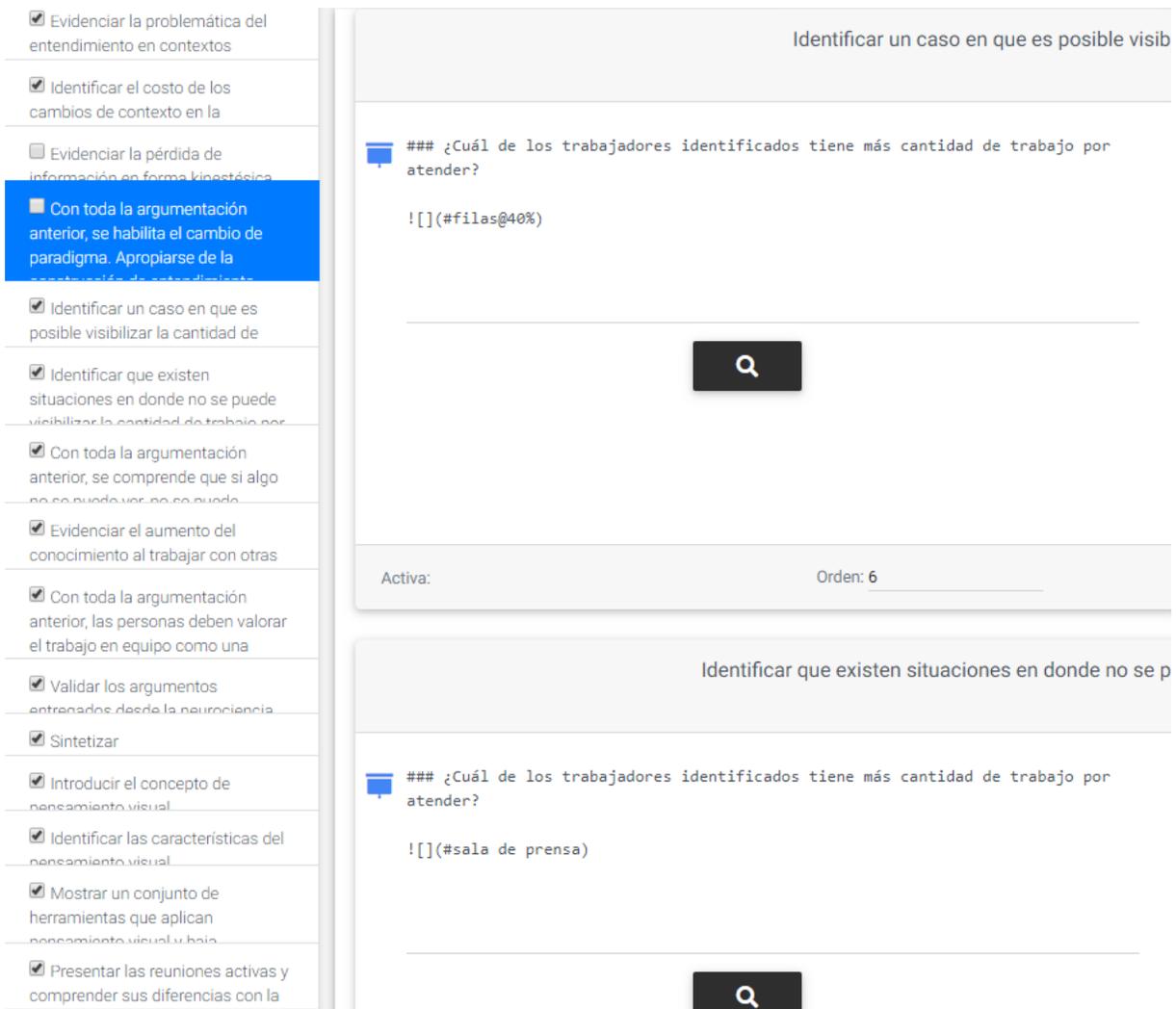


Figura 6.5: Lista de intenciones para navegar el contenido

nunca pierda de vista su ubicación actual en el módulo.

6.2.4. Previsualización

Ya que el editor, a diferencia de Microsoft Word o PowerPoint, éste no permite trabajar sobre el resultado exacto que se obtendrá, sino sobre una abstracción del documento. Es decir, no se puede esperar que el usuario imagine con exactitud el resultado de su implementación, por lo tanto, es necesario previsualizar de alguna manera la forma en que se lucirá el contenido implementado, como lo hacen los editores de LaTeX, como por ejemplo Overleaf³. Incluso teniendo una previsualización, se necesita poder navegar rápidamente sobre ella, según los lugares en los que se realicen cambios.

La solución implementa un botón de previsualización, que abre una ventana nueva con una previsualización del módulo completo, pero posicionado en la actividad asociada al botón que se presionó. Presionar el botón de previsualización de otra actividad no abrirá una ventana

³<https://www.overleaf.com/>

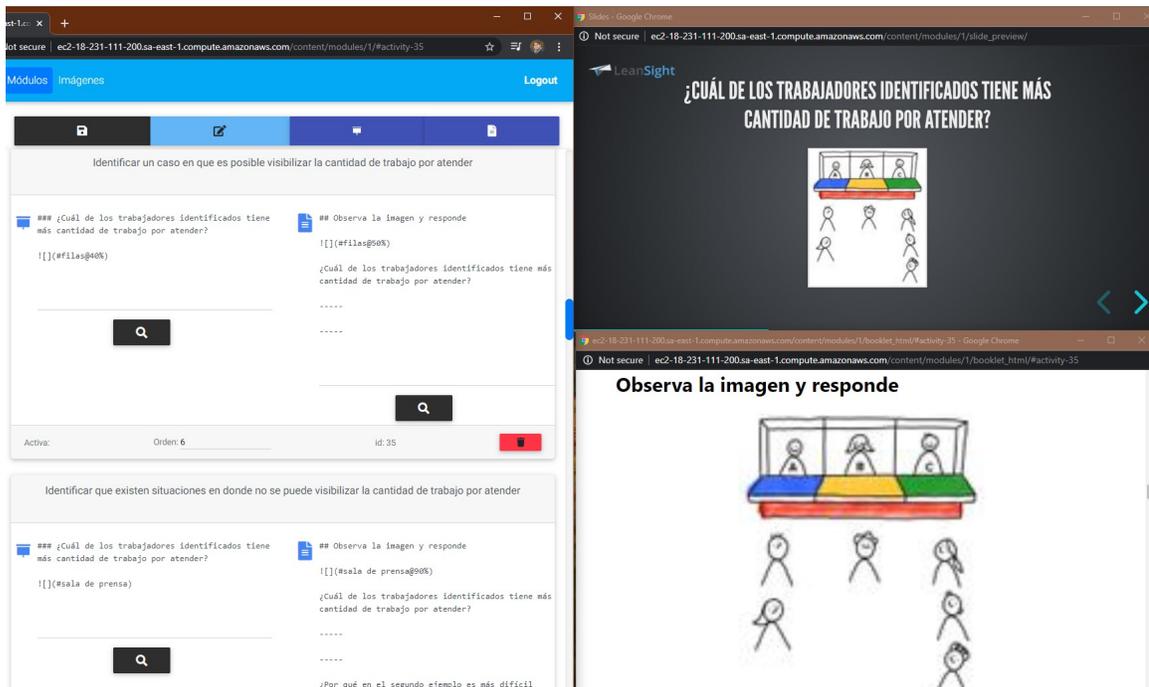


Figura 6.6: Ejemplo de previsualización en cuadernillo y diapositivas

nueva, sino que actualizará la posición a la actividad seleccionada. Salir de la vista de edición cerrará también todas las ventanas de previsualización para la comodidad del usuario.

En la Figura 6.6 se muestran en conjunto la implementación de la Figura 6.2 y sus previsualizaciones. Al presionar el botón negro con el ícono de lupa, se abre una ventana con la previsualización correspondiente y, si ya se encuentra abierta, se actualiza su posición a la actividad seleccionada. Esto habilita una navegación simple y ágil de la previsualización, reduciendo los obstáculos asociados a la implementación de documentos en un lenguaje de marcado como Markdown. Como se puede observar, las imágenes no tienen la resolución original dentro de la previsualización, en su lugar se usan versiones livianas de las imágenes generadas al momento de subirlas; de esta forma, la actualización de la previsualización es considerablemente más rápida, reduciendo los tiempos de carga de decenas de segundos (con imágenes grandes) a tiempos inferiores a dos segundos, buscando que el usuario no necesite esperar para ver el resultado de sus cambios.

6.2.5. Biblioteca de módulos

Cada módulo es único, y aunque pueden compartir actividades, cada uno persigue sus propios objetivos de aprendizaje. Es necesaria una visualización en que se puedan encontrar todos los módulos implementados, permitiendo agregar módulos nuevos y editar sus detalles.

Frecuentemente se accede a los módulos exclusivamente para exportarlos, por lo que es conveniente entregar la opción de exportar algún recurso sin tener que entrar al editor. La implementación cuenta con botones que permiten realizar las acciones relevantes para la administración de módulos.

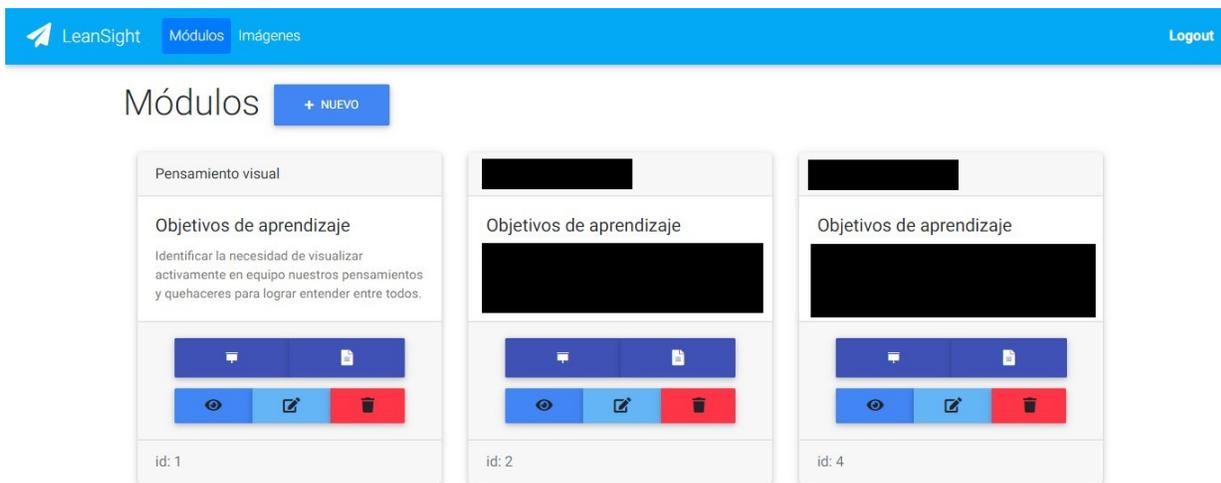


Figura 6.7: Biblioteca de módulos

Como se muestra en la Figura 6.7, cada módulo se representa como una tarjeta con su título y descripción, además de un botón para cada acción que se puede realizar: (1) los dos botones superiores son para exportar respectivamente la presentación de diapositivas o el cuadernillo, así no es necesario entrar a la implementación siendo un facilitador que solo busca los recursos; (2) el botón inferior de la izquierda permite entrar a la vista de edición del módulo y permite ver la implementación del mismo; (3) el botón inferior central permite modificar los metadatos del módulo, su título y sus objetivos de aprendizaje, por último (3) el botón inferior derecho permite eliminar el módulo, pidiendo una confirmación después de presionarlo, por ser una acción irreversible.

6.2.6. Guardado automático

Como el editor está en una plataforma web, se espera que cerrar accidentalmente la ventana sin haber guardado los cambios sea un error común. Una primera solución a este problema es pedir una confirmación cada vez que se desee salir de la vista de edición. Además de lo anterior, una prueba de usabilidad demostró que, al presionar el botón de previsualización, el usuario espera que la previsualización mostrada refleje los últimos cambios, siendo necesario guardar los cambios y volver a cargar la ventana externa.

La solución implementa un guardado automático cada vez que se presiona un botón de previsualización y maneja automáticamente la actualización de la ventana externa. El guardado automático no recarga el editor, permitiéndole al usuario realizar cambios con alta frecuencia y ver los resultados inmediatamente, sin alterar su flujo de trabajo.

6.2.7. Notificaciones

Si el guardado es automático, es necesario mostrarle al usuario que los cambios se guardaron exitosamente o mostrar un mensaje de error si ocurrió. De esta manera, el usuario podrá tener confianza en que sus cambios no se perderán o estar consciente de los errores que ocurrieron en el proceso.

El éxito o fracaso del guardado se muestra en un globo de texto sutil que desaparece automáticamente después de unos segundos, como se ve en las Figuras 6.8 y 6.9. Adicionalmente, se indica si no se detectaron cambios por guardar, para evitar confusiones en ciertos casos de uso.



Figura 6.8: Notificación de éxito durante el guardado de cambios

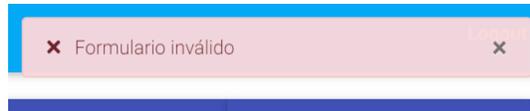


Figura 6.9: Notificación de fracaso durante el guardado de cambios

6.2.8. Administrador de imágenes

Una de las propagaciones de cambios más costosa es la que corresponde a cambios de imágenes. Una imagen que se encuentre en varios lugares de los documentos, repartidos entre diapositivas y documento de texto, requiere un proceso tedioso al reemplazarla, en que se deben recorrer los documentos completos; peor aún, si la imagen se utiliza en otros módulos, siempre queda erróneamente sin reemplazar.

Se implementa una biblioteca de imágenes, que pueden ser administradas con todas las operaciones CRUD, pidiendo confirmación para la eliminación de imágenes. Se permite actualizar el texto con que se identifica a cada imagen para su posterior uso, la imagen misma y una descripción opcional. En el momento en que se guarda la imagen en la base de datos, se crea también una versión liviana de la imagen, para mostrar en las previsualizaciones, optimizando su tiempo de carga y agilizando el flujo de trabajo para el usuario. El uso de las imágenes durante la edición se expone en la siguiente sección.

La Figura 6.10 muestra la interfaz asociada a la biblioteca de imágenes, con rectángulos negros cubriendo información importante para la empresa, dejando descubierta la primera tarjeta para visualizar el ejemplo. Las imágenes se muestran en tarjetas, con botones para sus acciones relevantes. La cantidad de tarjetas por fila se adapta según el ancho del contenedor, ofreciendo la misma experiencia para resoluciones radicalmente diferentes. En esta vista se utilizan las imágenes de baja resolución, para optimizar tiempos de carga; la imagen original se puede ver en su vista de edición. Además, se observa en la figura un botón de edición y un botón de eliminación del recurso, éste último con un diálogo de confirmación en caso de que sea presionado por error; presionar la imagen misma lleva también al menú de edición, donde se visualiza con mayor resolución y puede ser descargada.

6.2.9. Insertar imágenes

Almacenar y administrar imágenes solamente tiene sentido si se pueden utilizar en la implementación de actividades. La sintaxis debe ser simple y, sobre todo, legible.

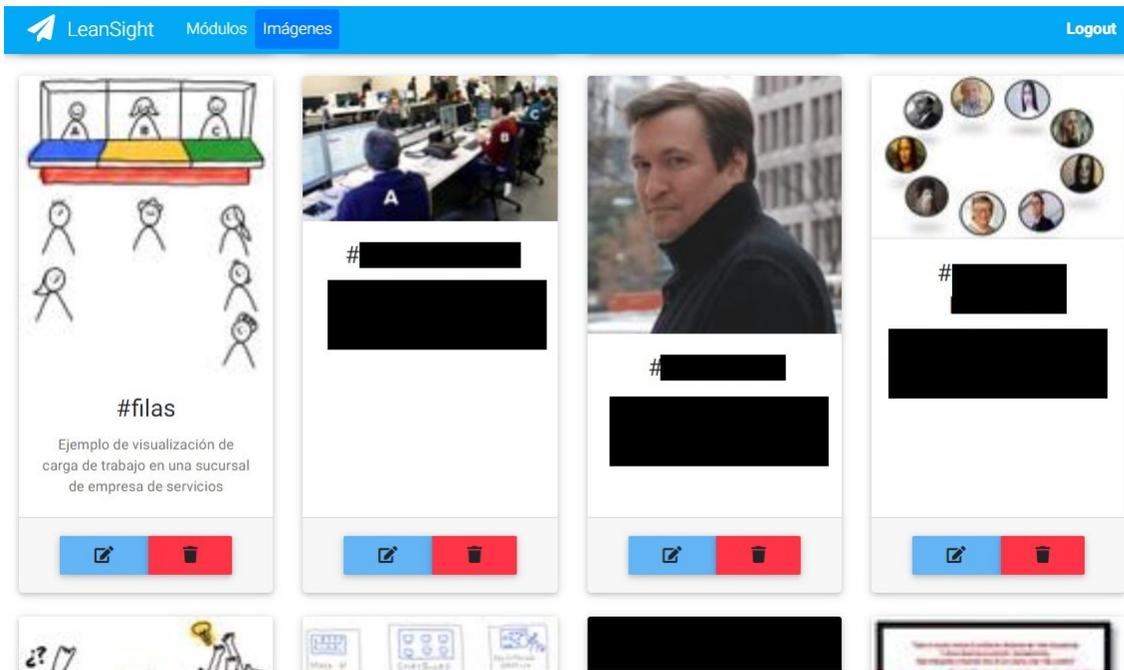


Figura 6.10: Biblioteca de imágenes

Se implementa una notación cómoda e intuitiva de inserción de imágenes, basada en la sintaxis de imágenes de Markdown y modificada con filtros de Pandoc. Se utiliza la misma sintaxis de Markdown, pero en vez de requerir una URL para la fuente de la imagen, se admite el uso de un identificador que le permita a la aplicación encontrar la imagen, o sea, el identificador que el usuario le dio a la imagen al subirla a la plataforma. No es necesario entonces mirar la previsualización para tener una idea de qué imágenes fueron insertadas.

La sintaxis de Markdown para modificar la visualización de las imágenes está basada en CSS, encontrándose fuera de la Zona de Desarrollo Próximo del usuario. Para simplificar el manejo de su tamaño, se extiende el filtro de inserción de imágenes; de manera que baste con escribir junto al identificador de la imagen el porcentaje del ancho del documento que se desea utilizar con la imagen. Entregando una idea tangible e intuitiva del significado del tamaño asignado.

Esta funcionalidad extiende la sintaxis de manejo imágenes y no interfiere con su uso original; sigue permitiéndose como fuente una URL a un sitio externo y notación de CSS para entregarle opciones adicionales de estilo.

```

 ### ¿Cuál de los trabajadores identificados tiene
más cantidad de trabajo por atender?



```

Figura 6.11: Ejemplo de inserción de una imagen bajo el identificador *filas*

En la Figura 6.11 se muestra un ejemplo de inserción de imágenes presente en las figuras anteriores. Se utiliza la sintaxis de imágenes de Markdown, pero con el identificador que

se usó para almacenar la imagen después de un símbolo # y la especificación del tamaño que se necesita en porcentaje. El software que administra imágenes se encarga de entregar la imagen correcta y en tamaño completo o de previsualización según corresponda, para optimizar tiempos de carga.

6.2.10. Reordenar actividades

En la situación anterior, no se visualizaban las actividades como elementos separados, lo cual hacía costosa cualquier operación CRUD sobre actividades. Las modificaciones sobre actividades son muy simples conceptualmente, pero siempre han tenido alta complejidad en su ejecución, una modificación típica en esta categoría es la de cambiar la ubicación de una actividad dentro del relato, que puede tomar varias decenas de minutos, incluso después de tener bien identificadas las actividades que se moverán.

Se ofrece una implementación simple para reducir el costo de una operación de este tipo, que es ciertamente perfectible, pero toma poco tiempo en su programación. Cada actividad tiene un campo numérico con su ubicación dentro del módulo, representando el campo que indica el orden de las actividades; modificar estos campos altera el orden de las actividades a voluntad. Al momento de guardar el orden nuevo, se normalizan los números, de manera que sean enteros consecutivos comenzando en 1, para evitar ordenamientos ininteligibles.

6.2.11. Seguridad

El acceso a la plataforma debe ser protegido, pues un usuario externo podría eliminar, sabotear o robar el cuerpo de contenidos que posee LeanSight. Se soluciona la problemática con el sistema de autenticación integrado en Django.

Por otro lado, Pandoc no replica las fuentes externas ni los archivos de estilo cuando se le entrega el archivo HTML como entrada, en su lugar, debe intentar convertir un sitio, recibiendo una URL como argumento. Como el ambiente en que se ejecuta Pandoc está afuera del sitio, necesita algún tipo de autenticación, de otra manera se crearía una brecha de seguridad para el contenido de LeanSight.

Al momento de solicitar el cuadernillo como documento PDF, se genera un token para el usuario autenticado y se ejecuta la instrucción de Pandoc con él. Así se mantiene el nivel de seguridad, incluso durante la generación de los archivos de salida.

6.2.12. Ocultar actividades

Los ajustes que se hacen a un módulo antes de usarlos en sesiones tienen casi siempre el objetivo de eliminar actividades para alcanzar los objetivos de aprendizaje planteados en un intervalo de tiempo restringido. En la situación anterior, estas modificaciones generan numerosos archivos con información duplicada, se pierden mejoras valiosas a ciertas actividades y no se documentan la ubicación ni el contenido de estos archivos.

Se ofrece la funcionalidad de ocultar actividades, simplificando la tediosa tarea que consistía en entrar a cada archivo, distinguir las actividades, decidir cuáles eliminar y recortar a mano con alta propensión al error. Con las actividades adecuadamente separadas, se puede

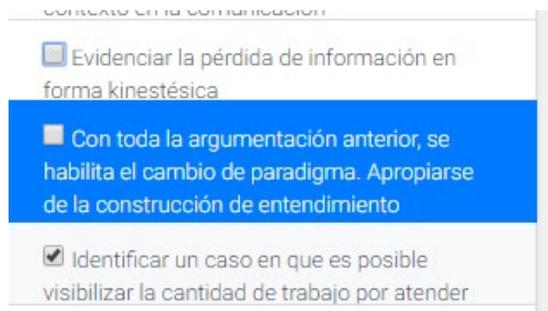


Figura 6.12: Ejemplo de actividades ocultas, la primera y segunda (seleccionadas) se encuentran ocultas y la tercera no

revisar cada una y decidir si debiese estar en la versión final según su intención. La aplicación permite ocultar la actividad con un clic, sin crear archivos separados ni eliminar la actividad del módulo.

Se muestra en la Figura 6.12 un ejemplo de actividades desactivadas, los efectos de esta acción se observan al exportar el documento, donde las actividades desactivadas no se encontrarán presentes.

6.3. Interfaz de usuario

Como muestra la sección anterior, varias funcionalidades se traducen en interfaces de usuario y aunque han sido explicadas por separado, es necesario visualizar las vistas completas para comprender la manera en que se manifiesta el flujo de trabajo. Las vistas de biblioteca ya han sido revisadas con suficiente detalle y solo queda mostrar en detalle la vista más compleja, la vista de edición.



Figura 6.13: Interfaz de edición completa

En la figura 6.13 se muestran la interfaz de usuario para la vista de edición, con ocho áreas

resaltadas, cada área y sus funciones se detallan a continuación, según el número con que se identifican en la figura.

1. *Navegación en la aplicación.* Corresponde a una barra de navegación siempre visible que permite navegar directamente hasta la vista de biblioteca de módulos o la biblioteca de imágenes. Como estos botones pueden ser presionados por accidente y salir de la vista sin guardar los cambios podría implicar volver a hacer un trabajo de horas, se muestra un diálogo de confirmación de redireccionamiento antes de salir de esta vista. Esto ocurre si se intenta abandonar la vista de edición de cualquier forma, incluso cerrando el navegador.
2. *Navegación en el módulo.* Una de las tareas más costosas para un diseñador antes de empezar a realizar modificaciones, es entender dónde empiezan y terminan actividades o el lugar en que se encuentra una actividad específica. Esta lista de actividades se encuentra siempre presente y resalta automáticamente la actividad que se esté visualizando, de esta forma el diseñador siempre sabe en qué lugar se encuentra y puede moverse rápidamente a lo largo del módulo.
3. *Acciones clave.* La primera acción clave es el guardado manual de los cambios, recordando que los cambios se guardan automáticamente al solicitar una previsualización o al presionar el atajo de teclado Ctrl+S. El resto de los botones también están presentes en la vista de biblioteca de módulos, con un estilo consistente; los tres botones cumplen las funciones de modificar los metadatos del módulo, exportar la presentación de diapositivas y exportar el cuadernillo respectivamente.
4. *Visualización de una actividad.* Cada actividad se muestra como una tarjeta que contiene todo lo necesario para editarla, en el encabezado la intención y en el cuerpo los editores de texto y las acciones relevantes. Esta sección cuenta con todas las actividades dispuestas verticalmente y a medida que el usuario se desplaza a lo largo de ella, la lista de actividades de la izquierda siempre resalta la primera actividad visible.
5. *Previsualización.* Cada actividad tiene un botón de previsualización por cada recurso, al presionarlo la ventana de previsualización se desplaza hasta la actividad seleccionada, o se abre si la ventana no estaba abierta. La previsualización se abre en una ventana nueva, porque el escenario típico en que un diseñador realiza modificaciones tiene dos pantallas conectadas al mismo computador.
6. *Ordenamiento de actividades.* Este campo permite cambiar la ubicación de las actividades dentro de un módulo, como se explicó en su funcionalidad asociada.
7. *Eliminación de una actividad.* Cada actividad puede ser eliminada con el botón de la esquina de cada una. Esta acción es irreversible, así que se muestra un diálogo de confirmación antes de concretar la eliminación.
8. *Edición de las visualizaciones.* Las visualizaciones se editan en campos de texto, con fuente monoespaciada para evitar errores por caracteres poco visibles y con la sintaxis basada en Markdown, que fue extendida para esta aplicación.

Capítulo 7

Validación de la solución

En este capítulo se estudia el impacto de la implementación, verificando que se hayan alcanzado los objetivos planteados y se dé solución a las problemáticas enunciadas. Como se ha presentado en otras secciones del documento, se obtuvo retroalimentación del equipo completo de LeanSight después de cada semana de desarrollo con casos de uso realistas y permitiendo interactuar con la aplicación sin ayuda. Además de las experimentaciones frecuentes, la validación culmina con una prueba realizada con un módulo real, una medición de impacto sobre los tiempos requeridos para cada etapa durante tal proceso, y la utilidad que los usuarios percibieron sobre la aplicación.

7.1. Ciclos de retroalimentación

Los miembros del equipo participaron activamente en los ciclos de retroalimentación cada una semana de desarrollo, mientras las condiciones lo permitieran. Entre los miembros del equipo de trabajo se encuentran otros Ingenieros en Computación, que pueden complementar las observaciones desde su perspectiva técnica. En general, los contenidos de los talleres que se imparten en LeanSight contienen estrategias de exploración y retrospectiva, por lo que el equipo ya cuenta con una mentalidad que facilita la experimentación y la mejora continua.

Durante los ciclos de retroalimentación se mantenía foco en que las principales problemáticas en torno a la modificación de recursos fueran resueltas y que la aplicación no indujera a errores serios durante su uso. Al mismo tiempo se mantenía especial atención sobre cuán intuitiva es la interfaz, dejando al equipo explorar la aplicación y hacer las preguntas que se consideren relevantes.

La solución como se presenta en este documento ya tiene incorporadas las observaciones obtenidas en estas sesiones de retroalimentación, de manera que la validación final (con un caso de prueba completo) representa un bajo riesgo para el proyecto, es decir, es poco probable que se descubra que la aplicación no ataque el problema correcto o que lo haga de manera incorrecta.

7.2. Prototipo de baja tecnología

Antes de iniciar el desarrollo de la solución, se aseguró que la lógica propuesta (Sección 4.2) tuviera sentido para el equipo, pues la solución se construye sobre ella. Con esto en mente, se realizó un prototipo de solución en papel durante el flujo de trabajo normal, buscando validar la lógica de capas.

Un ejemplo de los prototipos realizados se puede ver en la Figura 7.1 y consiste en un pliego de papel grande, donde se escribe cada intención educativa en notas autoadhesivas, creando un relato con la columna de notas. Hacia la derecha se escriben las implementaciones de cada actividad sin mucho detalle, solamente con lo necesario para entender la referencia, utilizando notas de distinto color al de las intenciones, para facilitar la comprensión del diagrama.

La flexibilidad que entregan las notas autoadhesivas entregó, además de un mejor entendimiento de las secciones de los módulos, un alto nivel de mantenibilidad. Se observa que el equipo puede adquirir fácilmente el hábito de trabajar sobre el relato en lugar de las visualizaciones, pues comprende el beneficio de mantener el alineamiento de cada implementación con su objetivo de aprendizaje, cada cambio a realizar se encontraba localizado en una actividad específica y navegar los contenidos se volvió inmediatamente mucho más simple.

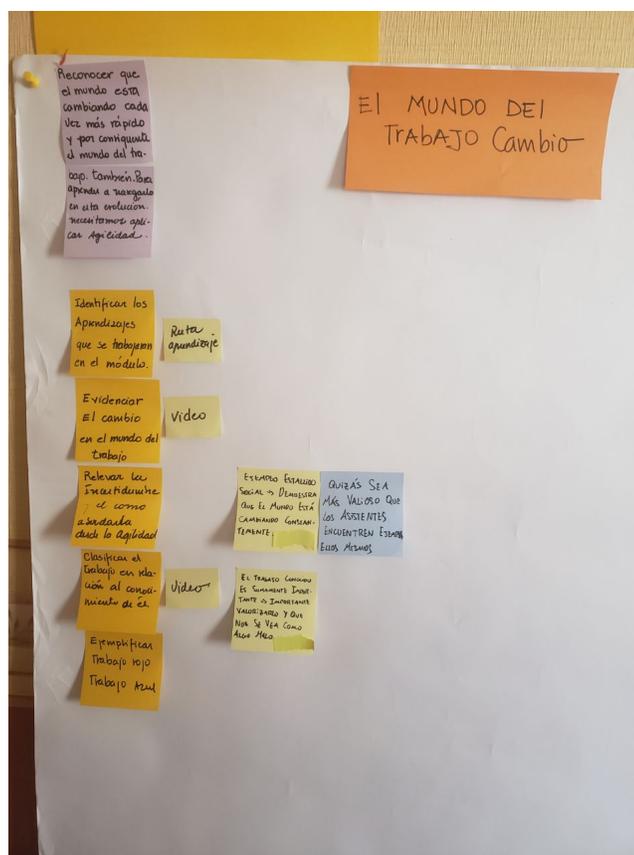


Figura 7.1: Primeras intenciones de uno de los prototipos realizados

Después de varios prototipos utilizados en sesiones de modificación de recursos, se comprueba que la lógica ayuda al equipo a gestionar de mejor manera la evolución del contenido educativo, entregando un modelo legible, flexible y accionable.

7.3. Prueba realizada

La solución se puso a prueba con un módulo educativo real, en particular se implementó el módulo que se encuentra en mayor estado de madurez según el equipo, el cual se titula: “Visualizar para entender entre todos”. Este es uno de los módulos que más ha pasado por modificaciones de eliminación de actividades para el ajuste al tiempo dado, tanto así que el equipo está en la búsqueda de su versión “módulo mínimo”, que corresponde a lo más corto que el módulo puede ser, pero sin perder consistencia educativa. Éste es también un módulo que introduce conceptos necesarios para otros módulos, entonces es normal que se encuentre en alguna sesión de casi cualquier cliente, por lo que recibe modificaciones con muy alta frecuencia.

La implementación de este módulo no fue posterior a la implementación de todas las funcionalidades expuestas, como esta sección podría insinuar, sino que se empezó a implementar tempranamente para obtener retroalimentación con anticipación. Las pruebas realizadas dieron lugar a varias funcionalidades y a nuevas priorizaciones de éstas.

Para poner a prueba qué tan intuitiva es la interfaz, se le entregó el completo control de la aplicación a la diseñadora educativa que tenía menor alfabetización digital del equipo, sin ayuda del desarrollador, pero sí con su supervisión, con el objetivo de rescatar observaciones sobre la usabilidad y posibles funcionalidades faltantes. Durante la prueba se midió el tiempo que tomaría realizar tareas comunes, para alimentar las estimaciones de la Tabla 3 (Sección 7.4) y también se verificó que la curva de aprendizaje fuera adecuada para el equipo. Esta no era la primera vez que el usuario veía la aplicación, pues ya había aportado observaciones en los ciclos de retroalimentación, aunque sí era una de las personas que no la había usado antes, por lo que representa el mejor usuario de prueba posible en este contexto.

El objetivo de implementar un módulo consistía en crear una versión que pudiera utilizarse en un taller, siendo necesario manejar todas las funcionalidades de la aplicación, es a partir de esta prueba que se obtuvieron las capturas de pantalla mostradas en el detalle de las funcionalidades (Sección 6.2). De la observación de esta experiencia se diseñaron e implementaron varios ajustes a la interfaz con el objetivo de mejorar la experiencia, como ubicación y tamaño de botones y contenedores. Habiendo conversado ya la lógica de actividades propuesta en este documento, el usuario percibe que se corresponde de manera muy natural y de inmediato empieza a pensar en las ventajas concretas que esta lógica y su implementación traen.

Se descubrió que muchos patrones se repiten en los documentos. y su implementación no es simple actualmente en varios casos. Sin embargo, gracias a los filtros de Pandoc, se pueden diseñar atajos para cualquier sintaxis. El atajo más frecuente era el uso de tablas pequeñas, para comparar situaciones o paradigmas, y debería ser la prioridad para automatizar en el futuro. En cualquier caso, existe la alternativa de volver a pensar el diseño y cuestionar si los elementos complejos eran realmente necesarios, o importar diagramas como imágenes.

Volver a diseñar los documentos con esta nueva herramienta es una buena oportunidad para replantear los diversos diseños que fueron implementados en un apuro, y que nunca fueron refinados. La herramienta cuenta con menos flexibilidad que las usadas en la situación anterior en cuanto a posibilidades de diseño gráfico, sin embargo, el usuario observa la

Tareas	Frecuencia (veces al mes)	Tiempo anterior (min)	Tiempo nuevo (min)
Cambiar de orden una actividad	2	30 - 60	2
Reemplazar una imagen	4	10 - 120	5
Quitar actividades para reducir su duración	10	40 - 150	40
Exportar recursos	15	15	3
Verificar la propagación de cambios	10	20 - 60	5
Encontrar un módulo actualizado	20	10 - 120	3

Tabla 7.1: Tiempo estimado para ejecutar modificaciones típicas anterior a la implementación (Tiempo anterior) y posterior a la misma (Tiempo nuevo)

posibilidad de realizar prototipos rápidos y enfocarse en la evolución y edición del contenido.

Durante el proceso mismo del traspaso del contenido, iban naciendo ideas de mejora para las actividades ya implementadas. El usuario por sí solo fue capaz de ver que estos cambios no eran difíciles de implementar, y se verificó que el tiempo que algunos cambios requieren es drásticamente inferior al de la situación anterior. Por ejemplo, después de haber usado una misma fotografía en varias actividades, el equipo sugirió reemplazarla por una creada por el ilustrador del equipo; en ese caso, se reemplazó la imagen en la biblioteca de imágenes y ningún trabajo adicional fue necesario.

La diseñadora educacional que implementó el módulo se mostró satisfecha con la solución obtenida; su retroalimentación será detallada más adelante en el documento. La diseñadora declaró que el trabajo sobre la plataforma fue cómodo, y aunque está consciente del trabajo que implica aprender la sintaxis de Markdown, comprende los beneficios que conlleva cambiar el flujo de trabajo y empezar a trabajar sobre el gestor y editor de contenidos desarrollado.

7.4. Medición de impacto

Utilizando un caso práctico se mostró que la lógica y la implementación se corresponden adecuadamente, atacando la causa raíz que encontramos al determinar nuestro desafío focal, asegurando además que las inconsistencias educacionales se reducirán significativamente. Sin embargo, no se debe olvidar que la solución no será efectiva si los dolores o síntomas persisten después de implantar la solución.

En la Tabla 7.1 se presentan tareas que frecuentemente realiza un diseñador educacional. El tiempo que tomaba cada tarea fue medido como parte del diagnóstico de la situación actual, indicado en la tabla como “tiempo anterior” (Tabla 3.1), que representa a la vez la medición de las soluciones anteriores, por ser ya la mejor versión de la situación hasta el momento de la observación. Detalles sobre las diferencias con soluciones anteriores se encuentran presentes en la sección de Recursos de Documentación (Sección 4.3), donde se discuten los problemas de esas implementaciones y cómo abordarlas. Para tener una buena idea del impacto sobre cada dimensión, se considera que se hacen aproximadamente 4 talleres por semana, y al menos la mitad de ellos requieren modificaciones antes de ser utilizados. Estos resultados son satisfactorios para el equipo, y se considera una mejora importante

respecto a la situación anterior, tal como se esperaría de una solución desarrollada a medida. Por supuesto, la aplicación tiene sus desventajas respecto a la situación anterior, por haber transformado por completo el flujo de trabajo y las herramientas de software usadas; por ejemplo:

- Es necesario replantear el diseño de los módulos, dadas las restricciones de los motores generadores de documentos.
- Independiente de lo intuitiva que sea la herramienta, hay un innegable costo de aprendizaje que se deberá pagar a causa de la nueva sintaxis.
- Depende de Amazon Web Services y requiere una conexión a internet estable durante la edición.
- Los archivos de salida no son editables, impidiendo hacer ajustes manuales posteriores a la edición en la plataforma.

Otras desventajas menores serán abordadas en la sección Trabajo a Futuro (8.2), donde se plantean desde las funcionalidades que las resuelven, apoyándose en el conocimiento posterior a la implantación de la solución, como se presenta en este informe.

En la Figura 7.2 se revisa el flujo de modificaciones planteado en la Figura 3.2 (sección 3.2), agregando de qué manera la plataforma aborda cada problemática, y cómo reduce la probabilidad de errores o inconsistencias, además de disminuir los tiempos necesarios para cada tarea.

Se había observado antes que el flujo es particularmente complejo cuando se cometió algún error en la modificación anterior, por ejemplo, habiendo olvidado subir a la nube los recursos actualizados. En la figura se muestra la forma en que la plataforma, además de reducir el tiempo de ejecución de varias tareas, previene gran parte de los errores que en la situación anterior se cometen con frecuencia. De esa manera, ésta apoya la tarea del diseñador educacional en mantener la consistencia del contenido, además de mantenerlo en su mejor versión posible de acuerdo con los conocimientos del equipo.

7.5. Utilidad percibida

Este trabajo comienza desde un conjunto de problemáticas que solo se podían percibir a través de sus síntomas, por lo que el trabajo investigativo previo al desafío focal genera valor para el equipo. Esto será efectivo en la medida que el conocimiento ganado les haga comprender de mejor manera la situación actual, para así cuestionar sus prácticas y colaborar en la construcción de soluciones efectivas desde ideas concretas. Los miembros de LeanSight adoptaron rápidamente la lógica propuesta para talleres, porque no era realmente una nueva forma de ver los recursos educativos, sino una forma de aclarar en concreto el sentido de los quehaceres ya presentes en el equipo. De esta manera, la terminología propuesta empezó a usarse de inmediato, haciendo uso de la distinción entre “actividades” y “recursos”, e “intenciones” y “relato”.

Luego de estar todos de acuerdo en la nueva forma de comprender el cuerpo de contenidos y las modificaciones sobre él, la solución tenía una base firme desde la cual construirse, comprendida por hipótesis validadas y entendimiento compartido. Naturalmente, el equipo

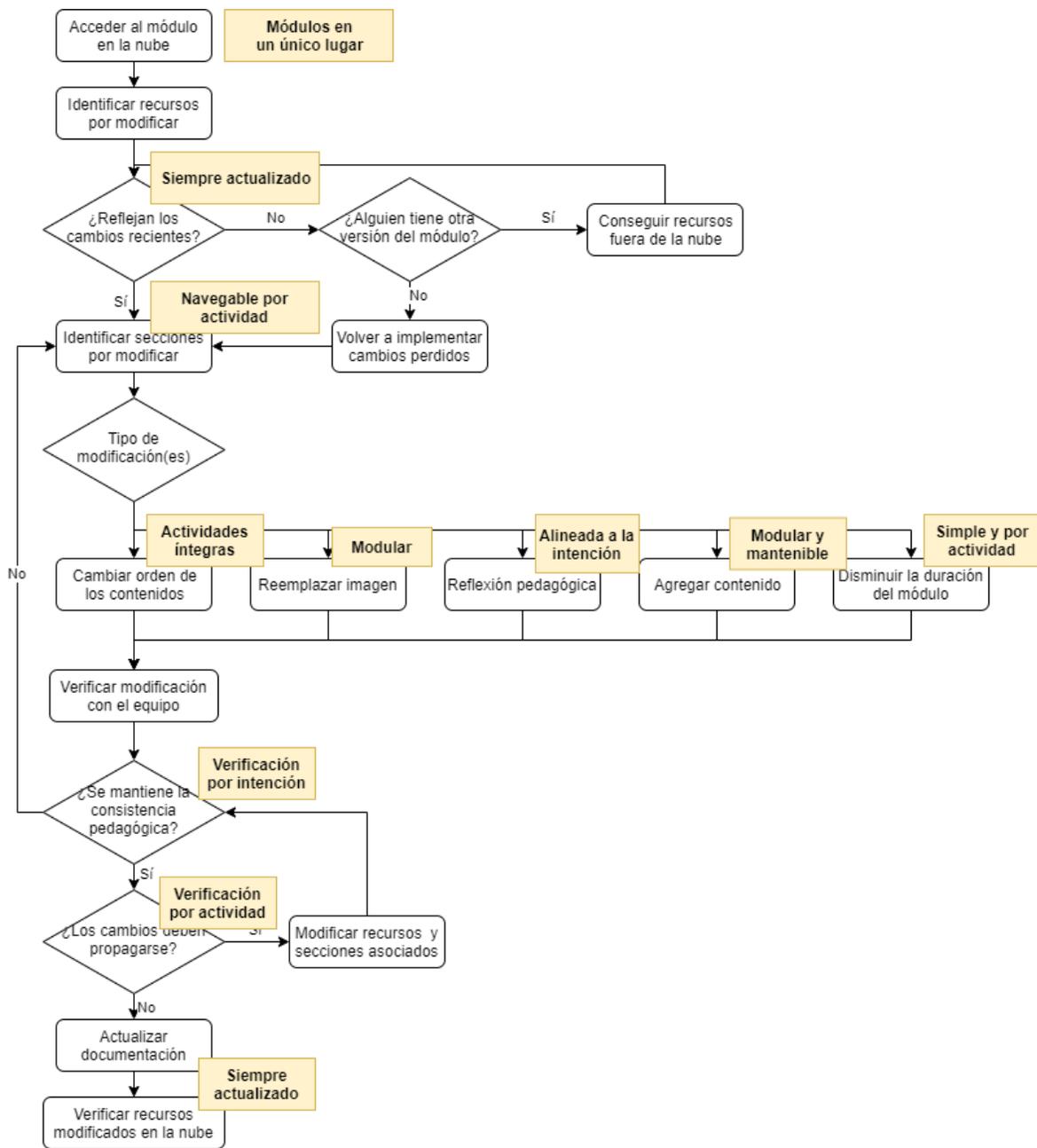


Figura 7.2: Flujo de modificaciones revisado con el impacto de la solución implementada

recibió de buena manera la aplicación, ya que pudieron rápidamente entender todas las tareas que se podían realizar en el software, y observar cómo cada una de esas tareas fue simplificada respecto de la situación anterior. El equipo aprueba la implementación en cada iteración, apoyando la priorización de funcionalidades y entregando retroalimentación temprana de usabilidad.

Los usuarios de la aplicación cuentan, en general, con un alto nivel de alfabetización digital y una alta capacidad de abstracción, que les permitiría aprender a utilizar la aplicación con gran facilidad. Sin embargo, la solución requiere una curva de aprendizaje adecuada, entonces la validación requiere mover el foco hacia las personas que tienen más dificultad en el uso de

herramientas tecnológicas.

Considerando lo anterior, la implementación de un módulo de prueba se realizó en conjunto con un miembro del equipo, especialista en pedagogía, que reconoce tener dificultad manejando herramientas computacionales. Su opinión después de las primeras horas de uso de la aplicación fue la siguiente:

“Los contenidos se muestran de forma ordenada e intuitiva. Cada actividad y su intención se puede encontrar y modificar con facilidad. Poder ocultar actividad por actividad también nos servirá para crear un cuerpo de contenido base y simplemente ocultar actividades y hacerlo calzar con el tiempo y los participantes. Esto sí o sí nos ordenará el trabajo.”

Con esto se termina de verificar que la recepción por parte del equipo fue positiva; aseguran que sus quehaceres serán más simples y cómodos, y que se eliminan obstáculos relevantes para ellos, permitiendo mantener el foco en el diseño de contenidos.

Capítulo 8

Conclusiones y trabajo a futuro

El trabajo tuvo como hito crítico el replanteamiento de la situación actual, que entrega una representación concreta de la lógica usada en los módulos implementados y un modelo de los flujos presentes en las tareas cotidianas relevantes para las problemáticas enunciadas.

El replanteamiento de la situación actual desemboca en un desafío focal, el cual complementa el objetivo general de desarrollar una herramienta de software que asista el proceso de construcción y mantención de talleres. La herramienta, acorde al desafío focal, debe crear una conexión sostenible entre las actividades educativas, sus intenciones y sus visualizaciones, habilitando mejoras futuras en las capas superiores, como la creación de patrones de relato mantenibles y objetivos de aprendizaje claros.

De lo anterior se desprende que los desafíos y objetivos se mantienen alineados entre sí, y también con la problemática observada inicialmente, entendiendo que, después de poner a prueba la implementación con un caso real, la recepción fue positiva por parte de los usuarios. Las mediciones verifican una mejora sustancial en la duración de las tareas de modificación más frecuentes, y también muestran que es posible extender la implementación para cubrir el resto de las problemáticas. Con esto se concluye que la implementación presenta un nivel de solución aceptable, que tiene una lógica intuitiva para el equipo, disminuye el tiempo requerido durante la edición de módulos, evita inconsistencias frecuentes y entrega un marco de trabajo para futuras mejoras en los procesos que se llevan a cabo dentro de la organización.

No se debe perder de vista que la solución diseñada tiene la desventaja de haber transformado por completo los procesos de modificación de módulos y seguramente existirá cierta resistencia al cambio, además de un costo asociado al aprendizaje del lenguaje de notación y la lógica subyacente. Además, es necesario rediseñar la visualización de todos los módulos, aunque no de los contenidos, para adaptarlos a las posibilidades que entrega el lenguaje y los motores generadores de documentos. Otras problemáticas observadas que no fueron cubiertas se exponen en la sección 8.2 (Trabajo a futuro), junto a una posible solución.

8.1. Lecciones aprendidas

El trabajo comenzó con una profunda inmersión en la problemática, con el objetivo de alcanzar un nivel de entendimiento adecuado para entregar una solución efectiva. Durante todo el proceso, inevitablemente se aventuran posibles soluciones, incluso entendiendo que carecen del entendimiento suficiente. En particular al término de este trabajo, se verificó en retrospectiva que los primeros conceptos de solución hacían sentido con los síntomas, pero se encontraban lejos de solucionar la problemática. Por ejemplo, pudo resultar fácil pensar en que separar los archivos por secciones y juntarlos con un programa al exportar daría solución al problema, pero solo complicaría la navegación durante los procesos de modificación, creando problemáticas más grandes que las que soluciona. El aprendizaje que se desprende es que un mal entendimiento de la problemática a solucionar no solo podría fallar en solucionarla, sino que también puede empeorar la situación, por lo que siempre deberían tenerse claras las hipótesis validadas y por validar antes de implementar cualquier contramedida sin un norte claro.

Por otro lado, las primeras demostraciones de avances al equipo cambiaron por completo las prioridades planteadas en un principio para el proyecto, y la ubicación y tamaño de distintos contenedores en la aplicación. Por ejemplo, entregar control de versiones sobre el contenido parecía prioritario e incluso crítico, sin embargo, se descubrió que otras funcionalidades eran mucho más importantes, como una representación modular de imágenes o la posibilidad de ocultar actividades al exportar los recursos. En resumen, se aprendió que la retroalimentación temprana, aunque podría traer cambios inesperados, asegurará que cada paso sea dado en la dirección correcta y se pueda entregar valor lo antes posible.

Finalmente, durante la realización de un trabajo de título en una empresa, con sus complejos problemas y prioridades, se debe manejar la dualidad entre plantear un proyecto, definir sus objetivos y posteriormente implementarlo, y mantener un flujo de valor para la organización mediante la observación de problemáticas y el planteamiento de prototipos que se puedan poner a prueba rápidamente. En particular, el enfoque de este documento, aunque se haga en retrospectiva, es predictivo en su articulación y es difícil corresponderlo con un flujo de trabajo de ciclos cortos y basado en experimentos. Se puede identificar una dualidad similar dentro del desarrollo de software en la relación simultánea con clientes y usuarios, ambos con enfoques diferentes y ambos relevantes al mismo tiempo. De esto y de las lecciones anteriores, se concluye que el foco debe estar siempre en la entrega temprana de valor y se deben alinear los objetivos en torno a esa visión, se entiende que situaciones similares son frecuentes en el mundo laboral y deben ser manejadas adecuada y oportunamente.

8.2. Trabajo a futuro

El trabajo realizado entrega solución a las problemáticas planteadas como desafío focal, sin embargo, quedan varias dimensiones de los quehaceres de LeanSight que pueden ser simplificadas o automatizadas.

A pesar de que las modificaciones se hacen típicamente en sesiones colaborativas, es importante poder visualizar en la plataforma la fecha de la última modificación, mejorando el entendimiento sobre los cambios ya incorporados al contenido.

Borrar todo el contenido de una actividad es un error que los usuarios podrían cometer; estos cambios no son reversibles y se tendría que volver a implementar la actividad. Esto no sería un problema si existiera la posibilidad de deshacer cambios o respaldar la información.

Si se cuenta con la funcionalidad antes mencionada, se tendría ya una forma primitiva de control de versiones. Implementar dicha funcionalidad no sería mucho más trabajo, y habilitaría un nivel suficiente de trazabilidad de cambios.

Implementar otras visualizaciones para las actividades es también un claro avance. La prioridad debería ponerse sobre las checklist de los módulos, las cuales están naturalmente asociadas a cada actividad y podrían generarse automáticamente al exportar el módulo. En ambos casos, la situación anterior y la actual las checklist están obsoletas, pero se reconoce su importancia y su complicidad con el modelo actual.

El trabajo colaborativo podría extenderse habilitando comentarios simples en actividades, aprovechando las cuentas de Django ya implementadas, para así permitir fácilmente iniciar discusiones sobre el contenido basado en actividades e intenciones.

La lógica de tres capas propuesta en este documento entrega claridad sobre gran parte de las problemáticas de LeanSight, más allá de los alcances de esta memoria. Abordar la capa del relato con los patrones de pedagogía y marketing que el equipo conoce, motivaría y mejoraría las discusiones sobre el contenido y la forma en que los participantes lo reciben.

La eliminación de una actividad o módulo es irreversible, y aunque se pida una confirmación, podría borrarse por error parte importante del contenido. Contar con funcionalidad que proteja al usuario de estos errores sería especialmente útil; por ejemplo, implementar una papelera que almacene los módulos, imágenes y actividades eliminados recientemente.

Durante las sesiones se usan muchas veces videos para motivar la reflexión y la inmersión en el problema. En la situación anterior se abre un reproductor de video sobre la presentación que se esté mostrando. Como ha sido discutido con el equipo, esto da una impresión poco profesional, pues se espera que los videos estén integrados a la presentación misma, y que el facilitador no necesite interactuar con el computador más allá del paso de las diapositivas. La funcionalidad de insertar videos en presentaciones supliría esta necesidad.

Se espera que la biblioteca de imágenes crezca rápidamente, complicando su manejo y navegación. Una implementación simple de filtros con un buscador o etiquetas entregaría al usuario una experiencia grata y libre de obstáculos.

Gran parte de las aseveraciones presentes en los talleres de LeanSight provienen de fuentes reconocidas, utilizadas como argumento por autoridad para convencer de manera efectiva sobre la importancia y correctitud del contenido. El manejo de estas fuentes por actividad, o bien la implementación de una biblioteca de fuentes, haría más simple la investigación futura y entregaría un proceso más ágil de evolución del contenido.

Algunos estilos se usarán frecuentemente, y aunque Markdown es bastante simple y legible, se preferiría el uso de plantillas en estos casos; es decir, formatos predeterminados que puedan ser llenados con información provista por el usuario.

Glosario

Con el objetivo de facilitar el entendimiento de los términos específicos del negocio, se entrega una breve definición de cada uno a continuación

1. *Módulo educativo*: Un conjunto de experiencias pedagógicas empaquetadas en una presentación de diapositivas, un cuadernillo y otros recursos para entregar habilidades y competencias durante un taller.
2. *Relato*: Estructura lógica de los talleres, que consiste en un conjunto de contenidos a entregar, actividades a realizar y consideraciones a tener.
3. *Actividad educativa*: Cada experiencia pedagógica por la que un participante pasa durante el estudio de los módulos, corresponde a la unidad fundamental de los módulos y cada una persigue su propia intención, alineada con el objetivo del módulo.
4. *Sesión*: El taller presencial, con los recursos y participantes que lo conforman.
5. *Diseñador educacional*: Persona experimentada en los contenidos que se quieren entregar y en conceptos pedagógicos. Crear sesiones y módulos educativos asegurando que se encuentren actualizados. Es el usuario de la solución presentada en este trabajo.
6. *Facilitador*: Rol que asume la persona a cargo de la ejecución de un taller, responsable de llevar a cabo cada actividad planificada y de evaluar cuidadosamente su efectividad. Los facilitadores también toman habitualmente el rol de diseñador educacional.
7. *Experto*: Rol que supervisa el alineamiento y cumplimiento de los objetivos de aprendizaje, debe ser una persona con vasta experiencia teórica y práctica en la experiencia que se desea entregar y sus aplicaciones.
8. *Participante*: Persona que asiste a un taller.
9. *Recurso*: Cada archivo o documento en que se apoyan diseñadores y facilitadores para llevar entregar habilidades y competencias. Ejemplos de recursos son el cuadernillo del participante y la guía del facilitador.
10. *Recurso documentación*: Recurso introducido en la sección 4.3 que busca documentar la función pedagógica de cada elemento en un módulo. Contiene la intención de cada actividad y los objetivos de aprendizaje.
11. *Cuadernillo del participante*: Documento impreso que se entrega a los participantes de un taller para que escriban sus observaciones y lleven a cabo actividades de reflexión.

Bibliografía

- [1] Anderson, L.W., Krathwohl, D.R., Airasian, P.W., Cruikshank, K.A., Mayer, R.E., Pintrich, P.R., Raths, J., & Wittrock, M.C. *A taxonomy for learning, teaching, and assessing: A revision of Bloom's Taxonomy of Educational Objectives (Complete edition)*. New York: Longman, 2001.
- [2] Puebla, L. *Gestión y automatización del proceso de cambio del cuerpo de contenido de LeanSight*. Memoria de Ingeniería Civil en Computación. Departamento de Ciencias de la Computación, FCFM, Universidad de Chile, 2019
- [3] Olivier Serrat (2009) *The Five Whys Technique*
<https://www.adb.org/sites/default/files/publication/27641/five-whys-technique.pdf> Última visita: Enero de 2020.
- [4] The Toyota GB Official Blog (2013). *Genba – Toyota Production System guide*.
<https://blog.toyota.co.uk/genba-toyota-production-system> Última visita: Enero de 2020.
- [5] Dubberly, H., Evenson, S. (2011). *Design as learning - The SECI Model*
<https://interactions.acm.org/archive/view/january-february-2011/design-as-learning-or-knowledge-creation-the-seci-model1>. Última visita: Enero de 2020.
- [6] McLeod, S.A. (2019). What Is the zone of proximal development? Simply Psychology.
<https://www.simplypsychology.org/Zone-of-Proximal-Development.html>. Última visita: Enero de 2020.
- [7] Iowa State University, Center for Excellence in Learning and Teaching. *Revised Bloom's Taxonomy*.
<https://www.celt.iastate.edu/teaching/effective-teaching-practices/revised-blooms-taxonomy/>. Última visita: Enero de 2020.