



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

GESTIÓN Y AUTOMATIZACIÓN DEL PROCESO DE CAMBIO DEL CUERPO DE
CONTENIDO DE LEANSIGHT

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN COMPUTACIÓN

LUCAS ANDRÉS PUEBLA SILVA

PROFESOR GUÍA:
JOCELYN SIMMONDS WAGEMANN

MIEMBROS DE LA COMISIÓN:
AIDAN HOGAN
JOSE PINO URTUBIA

SANTIAGO DE CHILE
2019

Resumen

LeanSight es una consultora en Innovación Organizacional y Tecnología que busca mejorar la cultura empresarial, de grandes organizaciones chilenas, del ámbito financiero, salud, previsional, entre otras. Esto se realiza a través de talleres o sesiones, para formarlos en Agilidad Organizacional, realizándose aproximadamente 30 talleres conformados de 6 módulos educativos, los cuales se componen de documentos interdependientes. Con el objetivo de mejorar la manera de impartir los conocimientos, los contenidos presentados durante los talleres son constantemente modificados, en promedio 2 a 3 veces por semana.

Si bien las modificaciones a los documentos usados en los talleres permiten mejorar el contenido impartido, estas provocan inconsistencias entre los distintos documentos, dada la posible propagación de un cambio a un documento asociado. Esto provoca trabajo suplementario al querer dictar una sesión, teniendo que verificar la consistencia entre los documentos de un mismo módulo, o buscando una versión consistente de entre las varias existentes. Además, las razones de las modificaciones no siempre se transparentan dentro del equipo, sin olvidar mencionar que no existe una manera de revisar los cambios a los documentos sin acceder a ellos, lo que consume tiempo de los colaboradores.

Previo a este trabajo, se utilizaba un sistema de control de versiones de documentos dentro de la empresa, el cual funciona solo para documentos individuales, por lo que no está adaptado para mantener la consistencia entre un conjunto de documentos, como lo son los documentos que componen los módulos de aprendizaje.

La solución propuesta es diseñar un sistema de control de versiones adaptado a la situación de la empresa. Para ello, el alumno propone un flujo de cambio basado en un ciclo de creación, uso, retroalimentación y modificación de los contenidos, para luego iterar sobre las etapas de uso, retroalimentación y modificación, con el objetivo de apoyar la creación y mejora del contenido de los talleres de manera sistematizada. Este flujo se basa en el funcionamiento de los *Sistemas de Control de Versiones*, además del funcionamiento de un *Issue Tracker*. Esto se realiza a través del desarrollo de varias herramientas de software que permiten por un lado hacer un seguimiento de las modificaciones realizadas, para luego asociarlas a las distintas instancias del contenido de los talleres, pudiendo diferenciar y acceder las distintas versiones de estos.

El trabajo realizado cumple con el objetivo de implementar un sistema de control de versiones adaptado a la situación de la empresa. Esto se ve reflejado en la capacidad de dicho sistema de gestionar y acceder documentos interrelacionados, además de permitir realizar el seguimiento de las modificaciones del contenido de los talleres, al poseer el 77,5% de los criterios de aceptación propuestos. No obstante, la usabilidad de las distintas herramientas desarrolladas presenta resultados mixtos, la gestión de documentos interrelacionados posee buena usabilidad, mientras que el seguimiento de las modificaciones tiene baja usabilidad. En otras palabras, es necesario seguir mejorando la experiencia usuario, pese a que el sistema posee gran parte de las funcionalidades necesarias.

Tabla de Contenido

1. Introducción	1
1.1. Contexto	1
1.2. Situación Actual	3
1.3. Problema a resolver	4
1.3.1. Síntomas del problema	4
1.3.2. Consideraciones adicionales	5
1.4. Solución propuesta	5
1.4.1. Objetivo general	5
1.4.2. Objetivos específicos	6
1.4.3. Relevancia de la solución	6
1.5. Metodología	6
1.6. Estructura del documento	7
2. Marco Teórico	9
2.1. Versionamiento	9
2.1.1. Version Control System o <i>VCS</i>	9
2.1.2. Git	10
2.1.3. GitFlow	11
2.1.4. GitHub	12
2.2. Tecnologías utilizadas	12
2.2.1. Google Drive	13
2.2.2. Google Apps Scripts	13
2.2.3. Visual Studio Code	14
2.2.4. Command Line Apps Script Projects - <i>CLASP</i>	14
2.3. Metodologías de trabajo	14
2.3.1. Ciclo de Deming	15
2.3.2. Nemawashi	15
2.3.3. Genba	16
2.3.4. 5 Por qué o 5 Whys	17
2.3.5. Análisis de causa raíz	18
2.3.6. Criterios de aceptación	18
2.3.7. System Usability Scale (SUS)	19
2.3.8. Producto semilla	20
2.3.9. Incrementabilidad de la solución	21
2.3.10. Entrega temprana de valor	21
2.4. Herramientas y flujos para poder manejar el cambio	22

2.4.1.	Funcionamiento de un Issue Tracker, o <i>IT</i>	22
2.4.2.	Discovery and Delivery	24
2.4.3.	Trello	25
2.5.	Resumen	25
3.	Determinación del Problema	26
3.1.	Situación Actual	26
3.1.1.	Actores	26
3.1.2.	Objetos	27
3.1.3.	Procesos	30
3.1.4.	Acotando el problema	35
3.2.	Delimitando los problemas	35
3.2.1.	¿Cómo es posible conocer la situación actual de un módulo?	35
3.2.2.	¿Qué información es necesaria para realizar modificaciones que permitan mejorar el contenido?	37
3.2.3.	¿De qué manera es posible comprobar el estado y propagación del cambio de los contenidos?	38
3.2.4.	¿De qué manera se puede mejorar el proceso de publicación de los recursos de una sesión o módulo?	38
3.2.5.	¿Cómo es posible saber qué cambió luego de una modificación a un contenido?	39
3.2.6.	¿De qué manera es posible diferenciar entre distintas instancias de un contenido?	39
3.2.7.	¿Cómo es posible mejorar el proceso de cambio?	40
3.3.	Discusión sobre la relevancia de la solución	41
3.4.	Características de calidad deseadas y/o requeridas por la solución	41
3.4.1.	Calidad de software	41
3.4.2.	Experiencia usuario	42
3.4.3.	Diseño iterativo de la solución	43
3.5.	Resumen	44
4.	Descripción de la solución	46
4.1.	Habilitando la caracterización del cambio	46
4.1.1.	Situación actual	46
4.1.2.	Objetivo del ciclo	47
4.1.3.	Alternativas de solución	47
4.1.4.	Diseño de la solución	47
4.1.5.	Implementación	48
4.1.6.	Validación	48
4.1.7.	Aprendizajes	51
4.2.	Caracterizando el cambio y su flujo	51
4.2.1.	Situación actual	51
4.2.2.	Objetivo del ciclo	52
4.2.3.	Alternativas de solución	52
4.2.4.	Diseño de la solución	52
4.2.5.	Alternativas de implementación	55
4.2.6.	Implementación	55

4.2.7.	Validación	59
4.2.8.	Aprendizajes	61
4.3.	Modelando los módulos	61
4.3.1.	Situación actual	61
4.3.2.	Objetivo del ciclo	62
4.3.3.	Alternativas de solución	62
4.3.4.	Diseño de la solución	62
4.3.5.	Alternativas de implementación	63
4.3.6.	Implementación	63
4.3.7.	Validación	65
4.3.8.	Aprendizajes	68
4.4.	Modelando el proceso de cambio	68
4.4.1.	Situación actual	68
4.4.2.	Objetivo del ciclo	68
4.4.3.	Alternativas de solución	70
4.4.4.	Diseño de la solución	70
4.4.5.	Implementación	70
4.4.6.	Validación	71
4.4.7.	Aprendizajes	73
4.5.	<i>Navegador</i>	73
4.5.1.	Situación actual	73
4.5.2.	Objetivo del ciclo	74
4.5.3.	Alternativas de solución	74
4.5.4.	Diseño de la solución	74
4.5.5.	Alternativas de implementación	75
4.5.6.	Implementación	76
4.5.7.	Validación	82
4.5.8.	Aprendizajes	82
4.6.	Resumen	83
5.	Validaciones	85
5.1.	Resumen de las validaciones intermedias	85
5.2.	Estudio de usabilidad sobre el uso del <i>Navegador</i> por el usuario Facilitador	87
5.2.1.	Diseño del estudio	87
5.2.2.	Ejecución del estudio	89
5.2.3.	Análisis de los resultados	91
5.2.4.	Diseño de la evaluación utilizando la System Usability Scale (<i>SUS</i>)	93
5.2.5.	Ejecución de la evaluación	93
5.2.6.	Análisis de los resultados	94
5.3.	Estudio de usabilidad sobre el uso del <i>MoME</i> y el <i>Issue Tracker</i> por el usuario Diseñador Educativo	95
5.3.1.	Diseño del estudio	95
5.3.2.	Ejecución del estudio	97
5.3.3.	Análisis de los resultados	99
5.3.4.	Diseño de la evaluación utilizando la System Usability Scale (<i>SUS</i>)	101
5.3.5.	Ejecución de la evaluación	101
5.3.6.	Análisis de los resultados	101

5.4. Resumen	102
6. Conclusiones	103
6.1. Resumen del trabajo	103
6.2. Trabajo futuro	106
6.3. Aprendizajes sobre el trabajo de título	107
Bibliografía	108
Apéndice	111
A. Criterios de Aceptación	112
B. Diagrama general detallado de la situación actual	117

Índice de Ilustraciones

2.1.	Diagrama del Ciclo de Deming, realizado por LeanSight.	16
2.2.	Diagrama explicativo del proceso de <i>Nemawashi</i> , realizado por Agustín Villena.	17
2.3.	Diagrama resumen de las relaciones entre los distintos elementos que conforman un Sistema de Seguimiento de Incidentes o <i>Issue Tracker</i> en inglés. . . .	23
3.1.	Diagrama simplificado de las interacciones entre los distintos actores.	29
3.2.	Diagrama simplificado de la relación entre los distintos procesos dentro de <i>LeanSight</i> , v20181211.	31
3.3.	Diagrama explicativo del proceso que busca generar entendimiento en los aprendices v20181211. El recuadro rojo indica una agrupación entre el contenido educativo, los medios y los experimentos.	33
3.4.	Diagrama explicativo del proceso que busca formar facilitadores v20181211. Los recuadros de colores indican distintas agrupaciones. En lo que respecta la flecha punteada, actúa de la misma manera que las flechas negras, pero sirve para diferenciar ambas flechas en caso de intersección.	34
3.5.	Diagrama explicativo del proceso de diseño y rediseño del <i>CdC</i> , también conocido como el proceso del cambio v20181211.	36
4.1.	Ejemplo de implementación de versiones del módulo <i>v20190304 ~ Visibilizar para entender entre todos</i> , en la esquina superior izquierda, la primera diapositiva de la presentación, en la esquina inferior izquierda, la última versión publicada, a la derecha, la primera hoja del cuadernillo; encerrado en rojo la ubicación de las versiones.	49
4.2.	Ejemplo de implementación de versiones del módulo <i>v190515.2146 Visibilizar para entender entre todos</i> , en la esquina superior izquierda, la primera diapositiva de la presentación, en la esquina inferior izquierda, la última versión publicada, a la derecha, la primera hoja del cuadernillo; encerrado en rojo la ubicación de las versiones.	51
4.3.	Diagrama resumen de los tipos de cambio y su propagación entre los distintos elementos del <i>CdC</i>	54
4.4.	Ejemplo de las plantillas para taller/sesión y módulo por utilizar en la bitácora de cambios realizada en <i>Google SpreadSheet</i> . Encerrado en verde está el elemento modificado, en magenta la razón del cambio, además los colores rojo y azul representan la implicancia de la propagación del cambio, en rojo en caso de propagación, azul cuando no la hay.	56

4.5. Ejemplo de las plantillas para medios e información por utilizar en la bitácora de cambios realizada en <i>Google SpreadSheet</i> . Encerrado en verde está el elemento modificado, en magenta la razón del cambio, además los colores rojo y azul representan la implicancia de la propagación del cambio, en rojo en caso de propagación, azul cuando no la hay.	57
4.6. Ejemplo de las plantillas para actividades por utilizar en la bitácora de cambios realizada en <i>Google SpreadSheet</i> . Encerrado en verde está el elemento modificado, en magenta la razón del cambio, además los colores rojo y azul representan la implicancia de la propagación del cambio, en rojo en caso de propagación, azul cuando no la hay.	57
4.7. Ejemplo de los pasos para comer una manzana reflejados en un tablero Kanban con todas sus componentes: encerradas en rojo las tarjetas o tareas; en azul, las columnas o estados; en verde, el límite de <i>WiP</i> ; en magenta, el orden de las tareas. La prioridad de las tareas por hacer está denotada por el orden de estas, más arriba es más prioritario.	58
4.8. Ejemplo del flujo seguido desde un Issue a un Cambio dentro de la implementación del <i>IT</i> en <i>Trello</i> , los recuadros del mismo color corresponden a ampliaciones de la captura de imagen, los colores verde y azul denota un cambio asociado distinto. Los colores rojo y morado, corresponde a el tipo incidente, en rojo issues y en morado cambios. Nótese que al haberse realizado ambos cambios, dado que se encuentran en el estado Done (recuadro morado), el issue también está en Issues - Done (recuadro rojo).	60
4.9. Comparación entre la foto con recuadros y una representación del estado final del Modelo de Módulos Educativos o <i>MoME</i> , luego de realizada la implementación para adaptar el modelo detallado de la situación actual a un módulo real, con recuadros que denotan los distintos elementos. Nótese que la misma imagen fue usada para las figuras 4.9(a) 4.9(b).	66
4.10. Implementación en <i>Google Spreadsheet</i> del <i>MoME</i> , a partir de los datos obtenidos en la actividad con baja tecnología representada en la figura 4.9. . . .	67
4.11. Captura de pantalla de la versión actual del <i>MoME</i> . Nótese que es posible observar todas las modificaciones realizadas a partir de las retroalimentaciones de los usuarios. Nótese que las figuras 4.11(a), 4.11(b) y 4.11(c) corresponden a la misma fila, pero por temas de espacio, se separaron en tres partes. Con respecto a la figura 4.11(d), corresponde a parte de la nomenclatura, además de ciertos detalles de implementación que facilitan la interacción entre los soportes del <i>MoME</i> e <i>Issue Tracker</i>	69
4.12. Diagrama de flujo del Proceso de Cambio a alto nivel	72
4.13. Captura de pantalla del contenido de un archivo de metadatos para el caso de un módulo.	77
4.14. Captura de pantalla del contenido de un archivo de metadatos para el caso de una sesión.	78
4.15. Captura de pantalla del contenido de un archivo de metadatos conteniendo la lista de sesiones.	78
4.16. Captura de pantalla del contenido de un archivo de metadatos conteniendo la lista de módulos.	79

4.17. Ejemplo de contenido de una carpeta de sesión, incluyendo la carpeta con el módulo, la carpeta “publish” con su contenido y el archivo de metadatos “.ls_session_metadata”.	80
4.18. Capturas de pantalla del <i>Navegador</i> inicial. En la figura 4.18(a) es posible ver una captura con los detalles del <i>Navegador</i> , mientras que en la figura 4.18(b) se ve el detalle de cómo se componen de manera abstracta.	81
4.19. Captura de pantalla del <i>Navegador</i> luego de las modificaciones realizadas a partir de la retroalimentación por los usuarios. Nótese que la manera de distribuirse el contenido es muy similar a aquél presentado en la figura 4.18, la gran diferencia reside en que hay menor información en pantalla, por lo que el usuario posee mayor facilidad para navegar los contenidos.	84
B.1. Diagrama general detallado de la situación actual v20181211	118

Capítulo 1

Introducción

Este capítulo pretende presentar brevemente el contexto dónde se desarrolla este trabajo de memoria. Para ello, se detalla la situación actual, desde la cual es posible abordar el problema y acto seguido, la propuesta de solución. Además se plantea la metodología que sigue el trabajo realizado para finalmente presentar la estructura del documento.

1.1. Contexto

LeanSight es una consultora en Innovación Organizacional y Tecnología que nace el año 2009, a partir de la experiencia de Agustín Villena aplicando metodologías ágiles en la Universidad de Chile [25]. Actualmente la empresa está compuesta por 8 colaboradores, contando al autor de este documento. Estos vienen de distintas áreas, como la Ingeniería de Software, Pedagogía, Diseño e Innovación y Creatividad.

De manera concreta, la empresa realiza talleres de metodologías ágiles para formar a equipos de clientes en una serie de principios que les permitan generar una nueva mentalidad, más adaptada al mundo del trabajo actual, dado que estas se rigen por paradigmas válidos en un mundo con menos tecnología que el presente. Los clientes son grandes organizaciones chilenas, del ámbito financiero, salud, previsional, etc. La cantidad de clientes fluctúa entre 5 a 10 clientes distintos por año, de mayor o menor magnitud.

Con respecto a los talleres, también llamados sesiones, estos se componen de uno o más módulos de aprendizaje. De los 6 módulos de aprendizaje existentes, 2 son introductorios y tienen una duración de aproximadamente un cuarto de jornada en total. Los 4 módulos restantes son de una duración de aproximadamente media jornada cada uno y tocan temas como el uso de la herramienta *Pensamiento Visual* [28], la Optimización de Flujos de Valor, formaciones en la metodología de *Tableros Kanban* [10] o Planificación Incremental, entre otros. Anualmente, se realizan aproximadamente 30 talleres, los cuales consisten de en promedio 6 módulos educacionales, por lo que se realizan cerca de 180 intervenciones de media jornada anuales.

Tal como se plantea en el párrafo anterior, los talleres están compuestos de módulos, los cuales representan unidades de conocimiento, y son realizados a partir de una serie de actividades pedagógicas. Estas actividades se apoyan en el uso de varios soportes, tales como una presentación, experimentos y una guía de alumnos o cuadernillo. Estos elementos son complementarios entre sí, y están diseñados para presentar la teoría y dar el espacio a los alumnos, conocidos como aprendices, para poner en práctica lo enseñado y apropiarse del contenido.

Ahora que el lector comprende a grandes rasgos el detalle de la labor de la empresa, es posible decir que su objetivo es el de mejorar la cultura empresarial a través de talleres de metodologías ágiles que entreguen una serie de conocimientos y acompañamientos en terreno, buscando generar un cambio de mentalidad en las empresas cliente.

Para lograr generar dicho cambio de mentalidad, *LeanSight* realiza talleres que tienen por objetivo situar a la audiencia en distintas situaciones hipotéticas que permitan resaltar la necesidad del cambio de paradigma, para luego presentar un principio que logre solucionar esta situación. No obstante, si bien los principios son entendidos dentro de la situación hipotética, generalmente el público no es capaz de aplicarlos a casos reales y en su cotidiano trabajar.

Dado lo anterior, gran parte del trabajo realizado en la empresa, y paralelo a la realización de los talleres, es buscar mejorar la manera de transmitir los conocimientos presentados durante estos últimos, a través de modificaciones a los contenidos de los módulos que los componen. De manera simplificada, luego de observar y determinar necesario el hecho de realizar un cambio, un elemento pasa por una etapa de modificación, luego de la cual debe ser validada y finalmente publicada.

Cabe resaltar que un alto porcentaje de las sesiones posee algún tipo de modificación con respecto a la anterior vez que se usaron sus contenidos. Se estima que 2 a 3 de las 5 formaciones realizadas semanalmente poseen algún tipo de modificación, realizado por alguno de los colaboradores. Si bien estos cambios pueden ser pequeños, como reemplazar una imagen por otra, existe la posibilidad de que este cambio se propague dentro del mismo soporte, o a algún otro soporte dada la interdependencia de estos. Lo anterior genera inconsistencias del contenido, además de requerir verificar la totalidad de los contenidos previo a su uso para una sesión. Esto dado que, si bien se posee un sistema de control de versiones, este no está adaptado a mantener la consistencia de un conjunto de archivos, por lo que no se adapta a la necesidad de la empresa.

A modo de resumen, el objetivo de la empresa es mejorar la cultura empresarial a través de talleres de metodologías ágiles, los cuales poseen unidades de contenido llamadas módulos. Los módulos a su vez están compuestos de una serie de actividades pedagógicas que se apoyan en varios documentos interrelacionados, buscando entregar a los alumnos o aprendices conocimientos específicos a través de una parte teórica acompañada de una parte práctica basada en una situación hipotética. Dada la dificultad para los alumnos de extrapolar los conocimientos, es necesario mejorar la manera de impartirlos, lo que implica modificaciones al contenido. Sin embargo, dado que el sistema de versionamiento actual no está adaptado a las necesidades de la empresa, esto genera una variedad de problemas, tales como inconsistencias en el contenido, o pérdida de tiempo al verificar si existen tales inconsistencias. A continuación

se presenta a grandes rasgos la situación actual, antes de abordar el problema.

1.2. Situación Actual

Para entender la situación actual, la cual es analizada en detalle en la sección 3.1, es necesario entender qué actores interactúan con qué elemento u objeto, dentro de qué proceso están estos últimos insertos, de qué manera se interactúa con ellos y por qué razón u objetivo se realiza esto.

Los actores son personas que interactúan con distintos objetos de distintas maneras, dentro de una variedad de procesos. Primero es necesario presentar al colaborador, es decir todo integrante de la compañía, independiente de su rol. Los roles que puede tomar, son de diseñador educacional, quién está a cargo de diseñar y modificar los contenidos educacionales de los módulos, o de facilitador, quién está a cargo de realizar las sesiones a los alumnos, o aprendices, es decir los clientes.

Los objetos son todos aquellos elementos con los cuales los actores interactúan. Existen varios tipos de objeto, los contenedores y los contenidos, donde los contenedores contienen los contenidos. Por otro lado, existen varias formas de objeto, aquellos explícitos e implícitos. Los objetos explícitos están totalmente definidos y existen, ya sea de manera virtual o física, mientras que los objetos implícitos están parcial o nulamente definidos, en otras palabras, se desconocen sus características.

Uno de los objetos más importantes es el *Cuerpo de Contenido* (de ahora en adelante *CdC*), el cual corresponde a la totalidad de los contenidos educativos de la empresa, los cuales se encuentran en el dominio de *Google Drive* de la empresa y de manera local en los computadores de los colaboradores. Los paradigmas se refieren al conjunto de normas establecidas dentro de la cultura empresarial actual, las cuales se busca cambiar a través de talleres o sesiones, compuestas de uno o más módulos educacionales, es decir, los contenidos. Estos contenidos se componen a su vez de actividades pedagógicas, que buscan transmitir una serie de aprendizajes. Las actividades pedagógicas se apoyan de una presentación, un cuadernillo, experimentos y medios. Los experimentos buscan poner a los aprendices en situaciones hipotéticas que los ayuden a evidenciar la necesidad de un cambio de paradigma.

Los procesos de la empresa, tanto de cara al cliente como de manera interna son principalmente 3, el primero consiste en generar entendimiento, a través de la vivencia de actividades pedagógicas. Como se dijo previamente, la necesidad de mejorar la manera de transmitir los conocimientos, implica modificaciones a los contenidos.

El segundo proceso importante, es el de formación de facilitadores, el cual radica en capacitar a los colaboradores para dictar un cierto módulo. Esto se realiza a través del manejo de contenido y de instancias de práctica frente a clientes, siempre acompañado de algún facilitador con mayor experiencia. Esto tiene por objetivo formar al equipo para poder sostener la carga de talleres semanales y tener mayor flexibilidad al momento de organizar la semana.

El tercer y último proceso se trata de la gestión del cambio, dada la búsqueda constante de

mejora de las actividades pedagógicas, y por ende la gran cantidad de cambios que se realizan a estas, se hace necesario gestionar dicho cambio. Si bien *Google Drive*, es decir el lugar dónde se almacena el *CdC*, permite la gestión del cambio de archivos singulares, el sistema no está adaptado para gestionar el cambio de múltiples documentos interrelacionados, como lo son los módulos. Este proceso presenta una oportunidad de mejora, al poder adaptar metodologías de gestión de versiones desde el área de las ciencias de la computación, y será el foco de este trabajo.

1.3. Problema a resolver

Tal y como se plantea en la sección previa, el problema por abordar es el de la gestión de cambios en los contenidos del *CdC*, es decir, es necesario gestionar las modificaciones del contenido de los talleres.

De manera concreta existen 4 diseñadores educacionales que son parte del proceso de modificación de los módulos. Cada módulo está compuesto por, una presentación en *Google Slides*; un cuadernillo o guía del alumno en *MS Word* y guardado en *Google Drive*; una guía del facilitador, que explica el relato de un módulo desde la perspectiva de módulos de aprendizaje y realizado en *Google Slides*; recursos audiovisuales o medios; instrucciones para realizar los experimentos; lista de chequeo con los materiales necesarios para ejecutar un taller.

Un taller común consiste de 6 módulos: 2 introductorios y 4 con los contenidos propios del taller. Cada módulo con contenido propio tiene una duración aproximada de media jornada.

1.3.1. Síntomas del problema

En la actualidad, el proceso de cambio se basa en el funcionamiento de la plataforma *Google Drive*, mediante la edición individual de los documentos que conforman un módulo. Esto resulta en una variedad de problemas, tales como:

- Es común que cuando un relator requiere usar la versión actualizada de los documentos de un módulo, encuentre inconsistencias notorias entre ellos. Por ejemplo, puede ser que los cambios realizados en las diapositivas no estén reflejados en el cuadernillo del alumno, o vice versa, lo que proyecta poca profesionalidad. En general, es costoso propagar los cambios que se realizan en un documento a los múltiples documentos relacionados, debido a que se hace de forma manual. Esto genera un gran trabajo de verificación y búsqueda por parte del relator para encontrar una versión consistente que pueda utilizar en los talleres.
- Los cambios realizados no quedan registrados, en particular, no se sabe cuál fue la razón de la modificación.
- *Google Drive* permite dos formas de sincronizar cambios, offline, a través de *Backup and Sync from Google* y online a través de la aplicación web. En la práctica, se ha observado

que existe un conflicto entre ambos sistemas, que provocaría que versiones anteriores de los documentos reemplacen versiones más actualizadas, provocando la pérdida de cambios.

1.3.2. Consideraciones adicionales

Los profesionales que participan del proceso de modificación vienen de distintos ámbitos, tales como Ingeniería de Software, Pedagogía, Creatividad y Diseño e Innovación. Por lo mismo, la capacidad de uso de las tecnología de cada profesional es variada. Es por ello que hay que tener en cuenta que la solución que se vaya a implementar debe poder ser utilizada por todos.

1.4. Solución propuesta

Muchos de los síntomas del problema explicados anteriormente son similares a los observados en la gestión del código fuente en un proyecto de software. A continuación se presentan estas similitudes, en la tabla 1.1:

Gestión del <i>CdC</i>	Gestión de código fuente
Módulo inconsistente	Release de software con bugs
Cambio sin razón	Cuando en el software no existe una correspondencia entre los cambios en el código y las historias de usuario, por lo tanto se pierde visibilidad y trazabilidad del estado de avance en el desarrollo
Conflicto Online - Offline	Concurrencia de modificaciones al código fuente, lo que provoca que se pierdan cambios

Tabla 1.1: Tabla de similitudes entre el problema de gestión del código fuente y la gestión de documentos del *CdC*

La solución propuesta es diseñar e implementar un sistema de control de versiones y el flujo de gestión del cambio asociado, para el conjunto de documentos que conforman un módulo de aprendizaje (multi-documentos).

1.4.1. Objetivo general

Esta memoria busca identificar, estructurar y proveer las herramientas de software para apoyar el proceso de gestión de cambios en los módulos de aprendizaje de *LeanSight*.

1.4.2. Objetivos específicos

1. Entender la situación actual a través de modelar el flujo de gestión de cambios de módulos de aprendizaje, de tal manera de poder identificar los problemas que este último posee y realizar un análisis de causa raíz a partir de ellos.
2. Diseñar, implementar y validar un flujo estructurado de gestión de cambios para módulos de aprendizaje compuesto de múltiples documentos.
3. Diseñar, implementar y validar un flujo estructurado que permita acceder a las distintas versiones de los módulos y/o las sesiones realizadas.
4. Apoyar el proceso de gestión de cambios con herramientas de software para minimizar la probabilidad de inconsistencia entre documentos, los cambios sin razón aparente y la pérdida de datos.
5. Proveer una buena experiencia usuario (*UX*) en el uso conjunto de herramientas, de tal manera de asegurarse que estas sean utilizadas.

1.4.3. Relevancia de la solución

En caso de lograr los distintos objetivos específicos, será posible transparentar a todo el equipo de diseñadores educacionales las razones y estados asociados a cada cambio, habilitando la colaboración a la hora de realizar modificaciones, y por ende mejorando la calidad del contenido de manera más eficiente en tiempo. Por otro lado, los facilitadores se verán beneficiados al poder realizar sesiones con módulos consistentes sin necesidad de perder tiempo verificando los documentos que las componen. En lo que respecta a los alumnos o aprendices, estos recibirán un mejor servicio, con contenido de mayor calidad y consistencia.

En otras palabras, el ciclo completo de uso, retroalimentación y modificación que siguen los módulos se acelera, lo que habilita a la larga crear mejor contenido y más rápido, asegurando una mejor experiencia para los clientes, y menos trabajo para los diseñadores educacionales y facilitadores.

1.5. Metodología

Este trabajo se basa principalmente en dos metodologías: la primera consiste en el ciclo de Deming [12], el cual se detalla en el capítulo 2.3.1, también conocido como ciclo de mejora, orientado a mejorar un servicio o producto a partir de iteraciones en pequeña escala. La segunda metodología es la del modelo de los *Tres Lenguajes* [24, 13], el cual plantea la existencia de tres lenguajes distintos: aquel asociado al problema, la solución y la implementación. Cada lenguaje posee vocabulario específico, y está relacionado con los otros dos. El objetivo es entender el problema, para luego diseñar una solución y finalmente llevar a cabo una implementación que permita resolver el problema planteado. Para ello, se realizan varios ciclos de mejora entre cada lenguaje, probando tanto alternativas de solución como alternativas de implementación.

Además, es necesario comprobar el funcionamiento de la solución implementada, para ello se realizan estudios de usabilidad de las distintas herramientas y su uso conjunto. Estas se basan en verificar si los usuarios son capaces de realizar una serie de tareas determinadas críticas del uso del sistema. Por tarea crítica, se refiere a una tarea necesaria para poder resolver el problema de fondo. Además, es necesario medir la usabilidad de dichas herramientas, para lo cual se utiliza la *Escala de Usabilidad de un Sistema*, o *System Usability Scale (SUS)* que permite entregar un índice de usabilidad de 0 a 100, y que determina si un sistema posee buena o mala usabilidad.

El conjunto de metodologías presentadas permite entender un problema real parcialmente conocido, para luego implementar soluciones que realmente lo resuelvan. Esto a través de la integración de los usuarios dentro de los ciclos de diseño e implementación, de tal manera de ajustar la solución a las necesidades de estos últimos, además de preocuparse de entregar una solución que vayan a ocupar y que sea usable.

En cuanto a lo que metodologías de desarrollo de software se refiere, se utilizan repositorios, control de versiones, control de calidad a través de revisión de pares, documentación del código, uso de ambientes de desarrollo adaptados a los lenguajes de programación utilizados, testing a partir de casos de usos, sin olvidar buenas prácticas de programación.

1.6. Estructura del documento

Con el objetivo de facilitar el entendimiento del lector, a continuación se presentará brevemente la estructura de este documento de memoria.

En el capítulo previo, se explica a grandes rasgos el contexto de la empresa, se detalla la situación actual, el problema abordado y la solución propuesta, enmarcadas dentro de la metodología utilizada.

En el capítulo dos, se busca plantear un contexto común el cual incluye todos los servicios o tecnologías externas que se utilizan para llevar a cabo la solución, además de explicar el funcionamiento de todas las metodologías utilizadas.

En el capítulo tres, se indaga en la situación actual para luego acotar el problema y abordarlo con mayor detalle, haciendo énfasis en los procesos involucrados y la relevancia de la solución. Se definen también los criterios de aceptación y características de calidad deseadas que debe cumplir la solución.

En el capítulo cuatro, se describen las iteraciones del *nemawashi* que componen la solución, en otras palabras, se expone la evolución de la comprensión de la situación actual y las distintas decisiones de diseño e implementación que permiten dar solución a los problemas planteados.

En el capítulo cinco, se aborda una validación final, la cual incluye estudios de usabilidad del uso de las soluciones implementadas, acompañada del uso de la *System Usability Scale (SUS)* que permite dar un índice de usabilidad del sistema.

Finalmente, el capítulo seis aborda las conclusiones del trabajo realizado, en particular de qué manera se logró qué objetivo. Se plantea además aquello que no se logró, para luego dar recomendaciones para trabajos futuros en el mismo proyecto y para proyectos similares. Finalmente se plantean a grandes rasgos los aprendizajes del alumno durante la realización de este trabajo.

Capítulo 2

Marco Teórico

En este capítulo, se busca proveer al lector del conocimiento teórico necesario para entender el resto del documento. Para ello, se presentan cuatro secciones distintas, la primera consiste en todo lo relacionado con el versionamiento, desde el funcionamiento de sistemas de control de versiones hasta software que lo implementa. La segunda categoría consta de las tecnologías utilizadas, desde ambientes de desarrollo a aplicaciones de software y librerías, se presentan las herramientas utilizadas para el desarrollo de este proyecto. La tercera categoría presenta las metodologías utilizadas para llevar a cabo la totalidad del trabajo, en otras palabras el marco metodológico que permite entender, diseñar, implementar y validar este trabajo. La cuarta y última categoría consiste en herramientas existentes para realizar seguimiento de incidentes, además de presentar un flujo de descubrimiento y entrega que será adaptado al contexto de la empresa, para finalmente presentar una herramienta que permite modelar flujos de manera sencilla.

2.1. Versionamiento

Esta sección pretende ahondar en aquello relacionado con el versionamiento, para ello se detalla el funcionamiento de un Sistema de Control de Versiones, seguido de la explicación de Git, software que lo implementa. Luego se expone uno de los flujos recomendados para utilizar Git, es decir el flujo GitFlow. Finalmente se describe a grandes rasgos las funcionalidades de la plataforma GitHub, la cual se basa en el funcionamiento de Git.

2.1.1. Version Control System o *VCS*

Un Sistema de Control de Versiones o *VCS* (de sus siglas en inglés) [3] es cualquier tipo de práctica que provee un seguimiento de los cambios o de las configuraciones de un producto.

Una de las particularidades de un *VCS* es la terminología utilizada. A continuación se presenta un breve resumen de los términos más relevantes:

- Repositorio: lugar donde se almacenan los datos e historial de cambios, a menudo en un servidor.
- Módulo: conjunto de directorios y/o archivos dentro del repositorio que pertenecen a un proyecto común.
- Revisión (*version*): versión determinada de la información que se gestiona. La última versión se suele identificar con el nombre **HEAD**.
- Abrir rama o ramificar (*branch*): se le llama a la acción de copiar un módulo en un tiempo determinado, ambas copias se denominan ramas, las cuales evolucionan de manera independiente. Existe la posibilidad de integrar (*merge*) dos ramas en una sola. El hecho de crear ramas permite probar funcionalidades en ambientes distintos. Si se determina que se quiere preservar la funcionalidad, entonces la rama que la posee es integrada a la rama principal.
- Desplegar (*checkout*): se refiere al hecho de crear una copia de trabajo local desde el repositorio. Es posible precisar la revisión, pese a que predeterminadamente se suele obtener la última revisión.
- Publicar o Enviar (*commit*): este sucede cuando una copia de los cambios hechos a una copia local es escrita o integrada sobre el repositorio.
- Conflictos: ocurren cuando el sistema no puede manejar adecuadamente cambios realizados por dos o más usuarios al mismo tiempo. Para ciertos casos específicos, el sistema será incapaz de fusionar los cambios, por lo que el usuario deberá resolver el conflicto combinando los cambios o eligiendo uno de ellos para descartar el otro.
- Resolver: es la intervención del usuario para atender el conflicto entre diferentes cambios a un mismo archivo.
- Integración o fusión (*merge*): se refiere a la unión entre dos conjuntos de cambios realizados sobre un archivo o conjunto de archivos en una revisión específica de dicho fichero o ficheros.
- Copia de trabajo (*workspace*): es la copia local de los archivos del repositorio, en un momento del tiempo o revisión específica.

De entre las varias razones que pueden existir para utilizar un *VCS*, se destacan la posibilidad de trabajar de manera colaborativa y de forma ordenada, lo que permite tener una mayor calidad del código y una mayor escalabilidad a la hora de desarrollar dicho código.

Existen distintas herramientas que permiten el control de versiones, entre ellas se destaca Git, y uno de los servidores que lo alojan GitHub, además de considerar el flujo de trabajo llamado GitFlow. A continuación se detallan tanto Git como GitFlow, dejando la explicación de GitHub para la subsección 2.1.4.

2.1.2. Git

Git [18] es un sistema distribuido de control de versiones desarrollado por Linus Torvald, pensado en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente. El propósito de este

sistema es poder coordinar el trabajo entre varios programadores, permitiendo seguir los cambios realizados a los distintos archivos de un proyecto.

Cada directorio de Git es un repositorio, con acceso a la totalidad del historial de cambios, capacidad de seguimiento de los cambios e independencia de conexión a un servidor o de un servidor centralizado. Este se estructura en forma de árbol, donde cada rama (*branch*) posee distintos nodos, los cuales representan cambios comprometidos (*commit*) por el usuario. Estas ramas pueden incorporarse (*merge*) lo que habilita el desarrollo en paralelo de distintas características (*features*) del código.

Entre las características principales del sistema se encuentran:

- Soporte para desarrollo no-lineal: se refiere a la capacidad de ramificar y mezclar las distintas ramas del código.
- Desarrollo distribuido: cada desarrollador posee una copia local del historial de desarrollo y de los cambios realizados a las distintas copias de los archivos.
- Compatibilidad con protocolos y sistemas existentes: los repositorios se pueden publicar a través de los protocolos HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol) o incluso SSH (Secure Shell), además de compatibilidad con otros sistemas de control de versiones como CVS y Subversion.
- Gestión eficiente de grandes proyectos: rápido y escalable.
- Autenticación criptográfica del historial: el identificar de un nodo depende del historial de cambios, por lo que alguna modificación al repositorio deja rastros.

La utilización de Git en este proyecto habilita el poder colaborar al desarrollar el código, además de poseer un historial de los cambios realizados.

2.1.3. GitFlow

GitFlow [5] es un flujo de trabajo definido para utilizar Git. Este pretende utilizar un número acotado de funcionalidades de Git para habilitar tener un proyecto ordenado de manera simple.

GitFlow se compone de los siguientes componentes, en forma de ramas o *branches*:

- Master: aquí se encuentra el código que va a pasar a producción. Cada vez que se incorpora código a esta rama, se tiene una nueva versión.
- Develop: corresponde a todos los cambios del código estables que conforman parte de la siguiente versión planificada del proyecto.
- Ramas auxiliares:
 - Feature: rama utilizada para desarrollar una nueva característica. Siempre surge de develop y es incorporada nuevamente a develop. Debe ser nombrada “feature-*<nombreCaracterística>*”
 - Hotfix: rama utilizada para arreglar errores del código en producción. Se originan de la rama master y se incorporan a la rama master o develop. Su nombre es de

la forma “hotfix-<nombreHotfix>”

- Release: rama utilizada para dar los últimos ajustes y arreglar bugs para preparar el código a su paso a producción. Se originan en la rama develop y se incorporan ya sea en la rama develop o master. Su nombre corresponde a la forma “release-<nombreRelease>”.

Al respetar las responsabilidades de estos distintos tipos de ramas, es posible tener un flujo ordenado a la hora de desarrollar utilizando el sistema de control de versiones de Git. Cabe resaltar que es necesario comprobar la estabilidad de la rama develop. Para ello, se verifica que las características o arreglos buscados por una rama feature o release se cumplen previo a la incorporación de esta con la rama develop. Se verifica además estabilidad y calidad del código. Este trabajo de verificación se realiza por un desarrollador que forme parte del proyecto, y se denomina Revisión de Pares o *Peer Review*.

GitFlow habilita entonces tener un código ordenado, estable y de calidad, a través del uso de pocas funcionalidades del sistema Git, por lo que es de simple uso para el usuario desarrollador.

2.1.4. GitHub

GitHub es una plataforma de desarrollo colaborativo para alojar proyectos de software que utiliza el control de versiones Git, es decir, permite gestionar el cambio que se realiza sobre un software o configuración de este. Además de las funcionalidades de Git, vistas en la subsección 2.1.2, se suman las siguientes características:

- Una documentación para cada proyecto.
- Página web para cada proyecto.
- Gráfico conteniendo información para los desarrolladores, sobre métricas u otro tipo de información relevante.
- Funcionalidades como si se tratase de una red social, orientada a la colaboración entre desarrolladores.
- Gestión de proyectos estilo Kanban.
- Posibilidad de solicitar que se integre una rama de código (*Pull Request*), con revisión de código por los pares y comentarios.

Esta plataforma es la segunda más utilizada con un 17 % del total del mercado [4], y contribuye a poder gestionar el código fuente de este proyecto de manera colaborativa, asegurando un estándar de calidad a través de la revisión de pares.

2.2. Tecnologías utilizadas

Esta sección apunta a describir en detalle las distintas tecnologías utilizadas. En primer lugar se habla de *Drive*, producto de Google donde se encuentra el *CdC*, para dar paso a

Google Apps Scripts y *Command Line Apps Script Projects* o *CLASP*, lenguaje de scripting que permite trabajar con productos de Google y la librería que permite trabajar de manera local en ellos. Finalmente se habla de *Visual Studio Code*, IDE desarrollado por Microsoft donde se desarrolla el trabajo realizado y que es compatible con las herramientas anteriores.

2.2.1. Google Drive

Es un servicio de alojamiento de archivos perteneciente a la compañía Google [9]. Posee, entre otras cosas, la capacidad de crear hojas de cálculo a través de *SpreadSheet*, presentaciones a través de *Slides*, documentos de texto a través de *Docs*, archivos que se guardan en la misma nube asociada a la cuenta de los usuarios.

Tal y como se plantea en el capítulo anterior, el *CdC* de la empresa se encuentra alojado en *Drive*, dado que éste proporciona el servicio de gestión de versiones de archivos singulares. En la misma sección, se expone la necesidad de poder gestionar versiones de una multitud de documentos relacionados, por lo que la plataforma no está adaptada a las necesidades de la empresa.

No obstante, es necesario basarse en el actual flujo de trabajo que considera el uso de *Drive*, para luego adaptarlo a las necesidades de la empresa. Es por ello que durante el transcurso de este proyecto, se sigue utilizando *Drive* como sistema de alojamiento.

2.2.2. Google Apps Scripts

Es un lenguaje de scripting para el desarrollo de aplicaciones ligeras en la plataforma *G Suite*, es decir, un servicio de Google que engloba muchos productos de la compañía, tales como *Docs*, *Slides*, *Sheets*, *Gmail*, *Calendar*, *Hangouts*, etc.

El objetivo de Google Apps Scripts [8] o *GAS*, es proporcionar formas fáciles de automatizar tareas en productos de Google y servicios de terceros.

Cabe resaltar que desde un script de *GAS* es posible implementar las funcionalidades en cualquiera de los productos de Google, además de aplicaciones web.

En particular, el *CdC* está alojado en *Drive*, y los distintos soportes utilizados se encuentran en *Slides*, *Docs*, *Sheets*, entre otros formatos. Es por ello, que al utilizar *GAS*, es posible automatizar ciertos procesos que involucren estos soportes y la plataforma. Ésto, a partir de APIs ¹ establecidas y bien documentadas.

¹Application Process Interface, es decir una interfaz que permite acceder a los servicios de una aplicación a partir de comandos específicos y generalmente simplificados.

2.2.3. Visual Studio Code

Visual Studio Code [14] es un editor de texto desarrollado por *Microsoft* para los sistemas operativos *Windows*, *MacOS* y *Linux*. Incluye, entre otras cosas: depuración de código, control integrado con Git, resaltado de sintáxis, finalización inteligente de código, etc.

Se utiliza Visual Studio Code dado que tiene soporte para el auto-completado de los tipos de *GAS*. Esto facilita el desarrollo de aplicaciones dentro del ambiente de *G Suite*. Además, el control integrado con Git, en particular GitHub, permite gestionar el código desarrollado utilizando repositorios locales y remotos.

Sin embargo, para poder utilizar el código, es necesario subirlo a la plataforma, para ello se utiliza la librería *CLASP*, que se presenta a continuación.

2.2.4. Command Line Apps Script Projects - *CLASP*

CLASP [7] es un proyecto de Google para poder conectar algún editor de texto local con los servicios de *GAS*. Esto permite desarrollar el código de manera colaborativa, utilizando buenas prácticas, en particular el uso de un repositorio para gestionar versiones del código.

A continuación se presentan los comandos básicos de *CLASP* brevemente, los cuales se usan a partir de la consola del editor de texto:

- *clasp push*: sube el código presente en el editor de texto al proyecto o script de *GAS* asociado.
- *clasp open*: abre el script de *GAS* asociado dentro del ambiente de *GAS* en el navegador por defecto.

A partir de estos comandos es posible desarrollar de manera local con Visual Studio Code, gestionar el versionamiento del código con GitHub, subirlo cuando sea necesario probarlo y/o implementarlo con *CLASP*. En otras palabras, conforman todos los servicios necesarios para poder desarrollar utilizando buenas prácticas y de manera colaborativa para el ambiente de *G Suite*.

2.3. Metodologías de trabajo

A continuación se presentan las distintas metodologías utilizadas para desarrollar este trabajo. En particular se describe el Ciclo de Deming, el cual permite entender el funcionamiento de Nemawashi y de las metodologías que integra, es decir *Genba* [11], *5 Por qué* [19, 21] y *Análisis de Causa Raíz* [27]. Por otro lado, se describe el funcionamiento de los criterios de aceptación [23], los cuales permiten determinar los comportamientos de la solución por implementar. Finalmente, se describe la manera de obtener el índice de usabilidad utilizando el *System Usability Scale* [2].

2.3.1. Ciclo de Deming

El Ciclo de Deming [12]², también conocido como *PDCA*, por las siglas de sus etapas, o *Espiral de Mejora Continua*, consta de 4 etapas:

1. Planificación (Plan): consiste en establecer todas las actividades necesarias para obtener el resultado esperado u objetivo. En otras palabras se requiere profundizar en el conocimiento del proceso; detallar las especificaciones de los resultados esperados; definir las actividades necesarias para conseguir el producto o servicio esperado, en función de los resultados especificados; establecer los objetivos y proceso necesarios para conseguir resultados necesarios de acuerdo con los requerimientos del cliente y la política organizacional.
2. Hacer (Do): Se realizan las actividades propuestas en el punto anterior. Dado que se trata de un ciclo de mejora continua, se podría considerar que las actividades propuestas consisten en aplicar algún tipo de mejora a algún producto o servicio. Se recomienda realizar ciclos de pequeña escala, y con un radio de acción acotado. Esto se explica en la etapa 4.
3. Verificar (Check): se analizan los resultados obtenidos y se contrastan con aquellos planteados en la etapa de planificación. El objetivo es determinar si se produjo la mejora planteada, validando o rechazando los supuestos que se pueden haber planteado en la etapa 1.
4. Ajustar o Actuar (Act): a partir de los resultados de la verificación en el paso 3, se procede a recopilar lo aprendido y ajustar los datos o supuestos iniciales, lo que ayuda a disminuir la incertidumbre sobre el proceso en sí. Es por ello que se recomienda hacer ciclos en pequeña escala y con un radio de acción acotado: mientras más ciclos, más información validada es posible recopilar, y al tener un radio de acción acotado, no impactan mayormente en la calidad del producto o servicio que se busca mejorar. Esto permite seguir realizando pruebas utilizando el mismo ciclo de Deming, por lo que también es llamada *Espiral de Mejora Continua*.

En la figura 2.1 es posible ver un diagrama de las 4 etapas planteadas.

2.3.2. Nemawashi

La metodología *Nemawashi* está basada en los pasos del ciclo de Deming aplicado a la gestión de cambio en una organización [20]. *Nemawashi*, palabra japonesa de significado literal “reordenar las raíces”, es el proceso mediante el cual se genera un consenso previo a proponer un cambio dentro de una empresa, o alternativamente tratar de refactorizar para hacer cambios de mejor manera. Concretamente, un maestro-líder, o aquella persona que posee más conocimiento, plantea un desafío a un responsable, quién está encargado de resolver dicho desafío. El responsable tiene por objetivo entender el problema para proponer contra-medidas

²El ciclo de Deming fue creado por Walter Andrew Shewhart y popularizado por Edward Deming en Japón durante la aplicación del Plan Marshall para la reconstrucción de dicho país luego de la Segunda Guerra Mundial



El ciclo de Deming

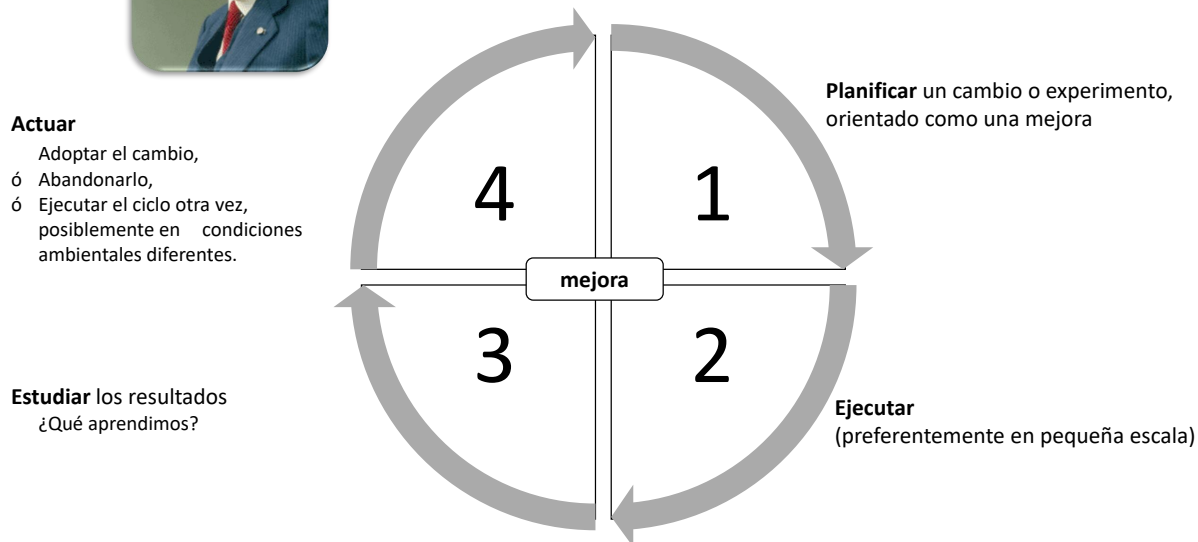


Figura 2.1: Diagrama del Ciclo de Deming, realizado por LeanSight.

que permitan dar solución al problema planteado. No obstante, las contra-medidas propuestas solo pueden ser llevadas a cabo cuando el responsable presenta un mayor entendimiento del problema que el maestro-líder, por lo que se sugiere experimentar en carne propia el problema y así generar *insights* al respecto. Finalmente, una vez aprobadas e implementadas las contra-medidas, toda nueva información es recopilada para ser entregada al maestro, quién propone un nuevo desafío, dando comienzo a una nueva iteración del mismo proceso.

Nemawashi utiliza tres metodologías de manera conjunta: *Genba*, *5 Por qué* y *Análisis de Causa Raíz*. A continuación se explican las 3 metodologías, haciendo énfasis en cuales son los datos de **entrada**, el **proceso** que siguen y finalmente los datos de **salida**. Esto permite luego ver de qué manera y en qué orden se conectan los distintos procesos involucrados.

2.3.3. Genba

El *Genba* [11] o *Gemba* [17]³, es un término japonés que significa “en el sitio de acción” o en la “escena del crimen”. Este concepto denota la necesidad de ir a vivir de primera mano una cierta experiencia, de tal manera de entender los procesos o problemas que pueda presentar esta última en carne propia. En otras palabras, se busca entender un proceso a partir de su vivencia, lo cual permite, ya sea confirmar o refutar hipótesis o juicios que uno pueda tener del proceso previo a experimentarlo.

³La romanización de los caracteres japoneses para la palabra “Genba” utiliza una letra “n”, pero al juntar la “n” con la sílaba “ba”, la pronunciación se dificulta en inglés, por lo que generalmente se pronuncia con una “m”. De ahí nace el término *Gemba*.

Nemawashi:

Cómo se realiza un cambio en la organización siguiendo la metodología

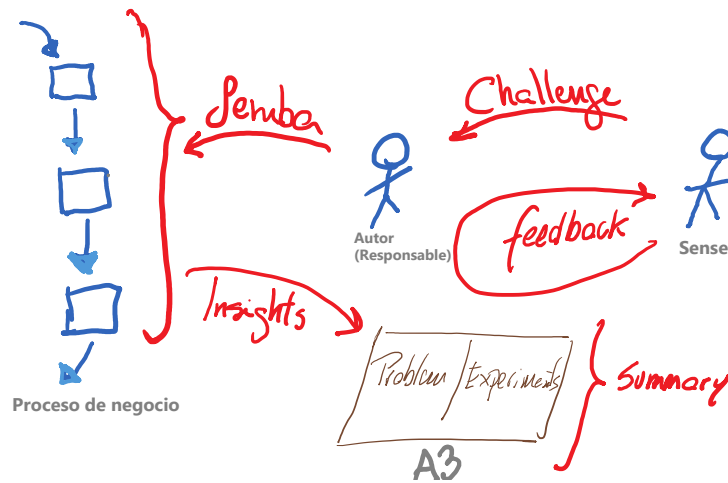


Figura 2.2: Diagrama explicativo del proceso de *Nemawashi*, realizado por Agustín Villena.

En este caso, la **entrada** es una experiencia cualquiera, el **proceso** es vivir dicha experiencia, la **salida** es el entendimiento de la experiencia, de los procesos involucrados en ella, además de los posibles problemas que pueda presentar, libres de prejuicios.

2.3.4. 5 Por qué o 5 Whys

Los *5 Por qué* [19, 21] es una técnica iterativa utilizada para explorar la relación causa-efecto de un problema en particular. El objetivo es determinar cuál es el problema subyacente al repetir la pregunta *¿Por qué?*, asociada a los posibles “síntomas” del problema.

Para dar a entender de mejor manera el proceso se plantea el siguiente ejemplo:

1. Un gerente recibe un informe: un trabajador se accidentó en el trabajo, ¿Por qué se accidentó?
2. El gerente va al lugar del accidente, en el cual hay una poza, con la cual se resbaló el trabajador. ¿Por qué hay una poza?
3. Luego de limpiar la poza, esta vuelve a aparecer al día siguiente ¿Por qué sigue habiendo una poza?
4. La poza era producto de una fuga en las cañerías, ¿Por qué las cañerías tienen fugas?
5. Luego de preguntarle al encargado de mantenimiento, este dice que la cañerías no han sido mantenidas ¿Por qué?
6. Finalmente, y luego de indagar lo suficiente (y aplicar los *5 Por qué*), el gerente encuentra la causa raíz del problema: no hay fondos suficientes para mantener el sistema de cañerías de la fábrica, por lo que se hace un ajuste de los presupuestos para solucionar

dicho problema. Nótese que de haberse resuelto los síntomas, es decir seguir limpiando la poza día tras día, nunca se habría solucionado realmente el problema y se hubiesen gastado recursos en ello.

Los 5 *Por qué*s toman como **entrada** una serie de problemas/síntomas, el **proceso** involucra plantearse la relación causa-efecto que podrían tener los distintos síntomas, de tal manera de generar entendimiento sobre los problemas subyacentes, para, finalmente, tener como **salida** la causa real del problema.

Cabe resaltar que es necesario tener criterio para evitar seguir preguntando “por qué” una vez que ya no aporta valor.

2.3.5. Análisis de causa raíz

El análisis de causa raíz (o ACR), es una metodología que tiene por objetivo identificar la causa raíz detrás de un problema, para poder darle solución, basandose en observaciones y relaciones causa-efecto [27]. Es importante distinguir entre los síntomas y el problema en sí, o raíz del problema: por ejemplo, un dolor de cabeza puede ser un síntoma de un resfriado, que vendría a ser la causa del problema.

El *ACR* consiste en las siguientes 5 etapas:

1. Definición del problema, qué es lo que sucede, cuáles son los síntomas.
2. Observación del problema, ¿qué prueba se tiene de su existencia? ¿Desde cuándo? ¿Qué impacto tiene? En este paso, es posible realizar *genba*, obteniendo así información sobre el problema.
3. Identificar posibles causas, en base a posibles relaciones causa-efecto, en este paso se utiliza la metodología de los 5 *Por qué*s
4. Identificar la causa raíz, ¿Qué efecto causal existe entre este problema y sus síntomas? ¿Cuál fue la verdadera razón detrás de lo ocurrido?
5. Diseñar e implementar soluciones al problema, de tal manera de evitar o al menos disminuir la probabilidad de que ocurra.

En otras palabras, el *ACR* toma como **entrada** un problema, el **proceso** consiste en observar síntomas a través de la vivencia de dicho problema, para luego indagar en las relaciones causa-efecto y poder llegar al problema de causa raíz, y finalmente tener como **salida** un diseño y/o implementación de solución a dicha causa.

2.3.6. Criterios de aceptación

Los criterios de aceptación de una solución son, “todas aquellas condiciones que la solución debe satisfacer para ser aceptada por un usuario, cliente o stakeholder” [23], entre otras cosas debe cumplir con al menos cuatro características:

- *Ser testeable, donde se valida o rechaza la solución de manera clara*: la respuesta debe ser binaria y permitir establecer si la solución cumple con los criterios o no.
- *Ser concisa y evitar ambigüedades*: debe ser entendible por la audiencia, mantenerse lo suficientemente precisa sin perderse en los detalles.
- *Basarse en un entendimiento común*: todos aquellos involucrados en el diseño y/o desarrollo de la solución debiesen poseer el mismo entendimiento, esto les permite saber qué se espera de la solución.
- *Denotar intención, no una solución*: hablar de una solución predispone a quien la desarrolla a un solo camino, la idea de un criterio de aceptación es denotar la capacidad de poder realizar algo, no determinar cómo se va a realizar. Por ejemplo si una persona tiene hambre, un criterio de aceptación para solucionar esto podría ser “darle comida”, no así “darle una manzana”, que si bien cumple con saciar su hambre, no es la única solución, se le podría dar una pera y obtener el mismo resultado.

Generalmente los Criterios de aceptación tiene la siguiente estructura:

Yo, como *Actor, rol y contexto,*
Cuando *Acción realizada,*
Entonces *Acción esperada.*

En donde las palabras en negrita son guías para saber qué se requiere a continuación, por lo que se dejan dentro del criterio mismo. Un ejemplo podría ser el siguiente:

Yo, como usuario de Google buscando informarme sobre “Felinos Extintos”,
cuando inicio la búsqueda en el sitio de Google,
entonces espero que se muestren 10 resultados sobre el tema “Felinos Extintos”.

Nótese que este ejemplo cumple con las cuatro características: es testeable, dado que se puede validar o rechazar; es conciso y evita ambigüedades: cualquier persona es capaz de entenderlo; se basa en un entendimiento común, si no se sabe lo que es Google, es difícil entender el criterio, sin embargo no es el caso; denota intención, no una solución: no se especifica ningún tipo de solución.

2.3.7. System Usability Scale (SUS)

El *SUS* es una manera de evaluar la usabilidad de un sistema que entrega un índice de usabilidad del 0 al 100 [2]. Para ello, se responde a 10 aseveraciones evaluadas con una escala del 1 al 5 que permiten denotar la percepción de los usuarios con respecto del sistema. Las aseveraciones son las siguientes:

1. Creo que usaría este sistema frecuentemente
2. Encuentro este sistema innecesariamente complejo
3. Creo que el sistema fue fácil de usar
4. Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar este sistema

5. Las funciones de este sistema están bien integradas
6. Creo que el sistema es muy inconsistente
7. Imagino que la mayoría de la gente aprendería a usar este sistema en forma muy rápida
8. Encuentro que el sistema es muy difícil de usar
9. Me siento confiado al usar este sistema
10. Necesité aprender muchas cosas antes de ser capaz de usar este sistema

De las 10 aseveraciones, 5 corresponden a aseveraciones positivas y 5 a aseveraciones negativas. Las aseveraciones positivas se encuentran en los números impares (1, 3, 5, 7, 9), y las aseveraciones negativas se encuentran en los números pares (2, 4, 6, 8 y 10). La fórmula para llegar al índice de usabilidad es la siguiente:

$$I_{usabilidad} = (\sum (P_{positiva} - 1) + \sum (5 - P_{negativa})) * 2,5$$

Dónde $P_{positiva}$ corresponde al puntaje entre 1 y 5 de las aseveraciones positivas, $P_{negativa}$ corresponde al puntaje de las aseveraciones negativas e $I_{usabilidad}$ corresponde al índice de usabilidad calculado.

El puntaje asociado corresponde a valores entre 1 a 5, los cuales responden a las aseveraciones de la siguiente manera:

Puntaje	Significado
1	“Totalmente en desacuerdo”
2	“En desacuerdo”
3	“Neutral”
4	“De acuerdo”
5	“Totalmente de acuerdo”

Tabla 2.1: Tabla resumen de los distintos puntajes en la encuesta cualitativa y su significado.

Esta escala permite entregar un índice de usabilidad entre 0 y 100, dónde un puntaje inferior a 51 refleja que la usabilidad es insuficiente; un puntaje de 68 indica que el sistema posee una usabilidad aceptable; un puntaje superior o igual a 80 denota que el sistema posee buena usabilidad. Sin embargo, esta cifra es solo una estimación, y debiese ser ajustado a cada sistema en caso de tener varias respuestas.

2.3.8. Producto semilla

El producto semilla es aquél producto que soluciona el problema de la manera más sencilla, pero que además posee las características necesarias para escalar a la solución final. Es similar al concepto de Producto Mínimo Viable (PMV o MVP por sus siglas en inglés⁴), sumando el hecho de poseer las características para escalar.

A modo de ejemplo, si se tiene como objetivo desplazar a una persona a gran velocidad, se

⁴Minimum Viable Product

plantean las siguientes opciones de producto semilla: un skateboard o una bicicleta. Ambas opciones cumplen el objetivo de manera sencilla, es decir movilizarse.

Ahora bien, al escalar la solución, dado que ya se logra movilizar a la persona, se desearía poder movilarla a una mayor velocidad. Para ello, en el caso de la bicicleta, es posible pensar en instalar un motor, lo que daría una bicicleta a motor, y eso eventualmente podría escalar a una motocicleta, por lo que se trata de un buen producto semilla.

El caso del skateboard es distinto, es más difícil agregar un motor dado el menor tamaño de la estructura, por lo que, en este caso, son las características del producto las que no están adaptadas para seguir escalando, luego no es un buen producto semilla.

2.3.9. Incrementabilidad de la solución

El concepto de solución incremental está ligado a aquél de producto semilla dentro del marco del desarrollo iterativo. Un producto semilla posee todas las características para escalar a la solución final, pero va creciendo "por incrementos".

Sea un producto semilla en un determinado momento, este se encuentra en estado **A** con la característica X . Al realizar una nueva iteración este producto escala a un estado **B**, el cual posee las funcionalidades X e Y . Como es posible notar, el estado **B** posee las características del estado **A** además de una nueva funcionalidad Y . Esta manera de hacer "crecer el producto semilla" se llama desarrollo incremental.

Volviendo al ejemplo dado en el apartado anterior, la bicicleta sería el producto en el estado **A**, pudiendo desplazar a una persona (característica X), mientras que la bicicleta con motor sería el estado **B**, pudiendo desplazar a una persona a gran velocidad (característica Y). Eventualmente, el producto final podría ser una motocicleta, con espejos retrovisores, tapabarros, un asiento más cómodo para el conductor, una maleta, etc.

Para el caso de este proyecto, la incrementabilidad de la solución es relevante para permitir validar y posteriormente construir sobre dicha solución validada.

2.3.10. Entrega temprana de valor

Denota la capacidad de la solución de entregar valor a la empresa desde etapas tempranas de su implementación. En otras palabras, luego de cada ciclo de diseño, y en función del entendimiento en ese momento del problema, se prioriza implementar el incremento de solución que permita entregar el mayor valor a la empresa. En este caso, el valor se entiende como aprendizaje o entendimiento del problema, y a su vez como un aporte al proceso de cambio.

A modo de ejemplo, si se quisiera vender a través de una página web un cierto producto, se podrían considerar dos facetas: el diseño de la página web, y la funcionalidad de venta de esta. Si se implementa primero el diseño, no es posible vender el producto, por lo que no se entrega valor, a pesar de tener un sitio web llamativo. Por otro lado, si se implementa la

funcionalidad de venta, independientemente de la estética, si es posible vender productos, lo que genera valor a la empresa, en este caso ingresos.

2.4. Herramientas y flujos para poder manejar el cambio

Esta sección pretende indagar en el funcionamiento de un *Issue Tracker*, herramienta que permite hacer el seguimiento de incidentes y es utilizada como base para poder seguir las modificaciones realizadas a los contenidos del *CdC*. Por otro lado, se explica el funcionamiento de un flujo de descubrimiento y entrega (*Discovery and Delivery* en inglés) lo cual permite desglosar las modificaciones en función de su grado de incertidumbre. Finalmente se explica el funcionamiento de la herramienta *Trello*, la cual permite modelar flujos de manera sencilla y que permitirá implementar parte de la solución.

2.4.1. Funcionamiento de un Issue Tracker, o *IT*

Un Issue Tracker o Sistema de Seguimiento de Incidentes es un sistema informático que permite a todos los usuarios de un sistema visibilizar y generar entendimiento compartido con respecto a los incidentes que hayan sido reportados dentro de dicho sistema.

Un *IT* posee el funcionamiento presentado en la figura 2.3:

A continuación se da más detalle del funcionamiento del *IT*:

1. Observación de un incidente o síntoma, el cual necesita de un contexto compuesto por:
 - La instancia o versión en que ocurrió el problema.
 - Una breve descripción de lo ocurrido.
 - La severidad de lo ocurrido, de tal manera de poder priorizar luego. A modo de ejemplo, si hay un error a nivel estético, no es necesario arreglarlo de manera urgente, no así el caso en que el usuario no sea capaz de ingresar a su cuenta.
 - Datos suficientes para poder reproducir el error, con tal de poder verificar si este persiste o no.
 - En qué lugar preciso ocurrió el error.
2. Declaración de lo que no funciona, y para ello es necesario declarar también el funcionamiento esperado u objetivo del cambio.
3. Determinación del orden en que se realizan los cambios, dado que es probable que exista más de uno.
4. Visualización del estado de los cambios, esto para todo usuario encargado de seguirlos, en otras palabras generar un contexto de entendimiento compartido dentro del equipo.
5. Creación de un historial de cambios, el cual responde a la pregunta ¿Cómo sé qué cambió? Este está compuesto por información respecto de:
 - Fechas de llegada del incidente y resolución de este, en caso de que haya sido resuelto. De todas maneras permite entregar el cuándo se inició y el cuánto se

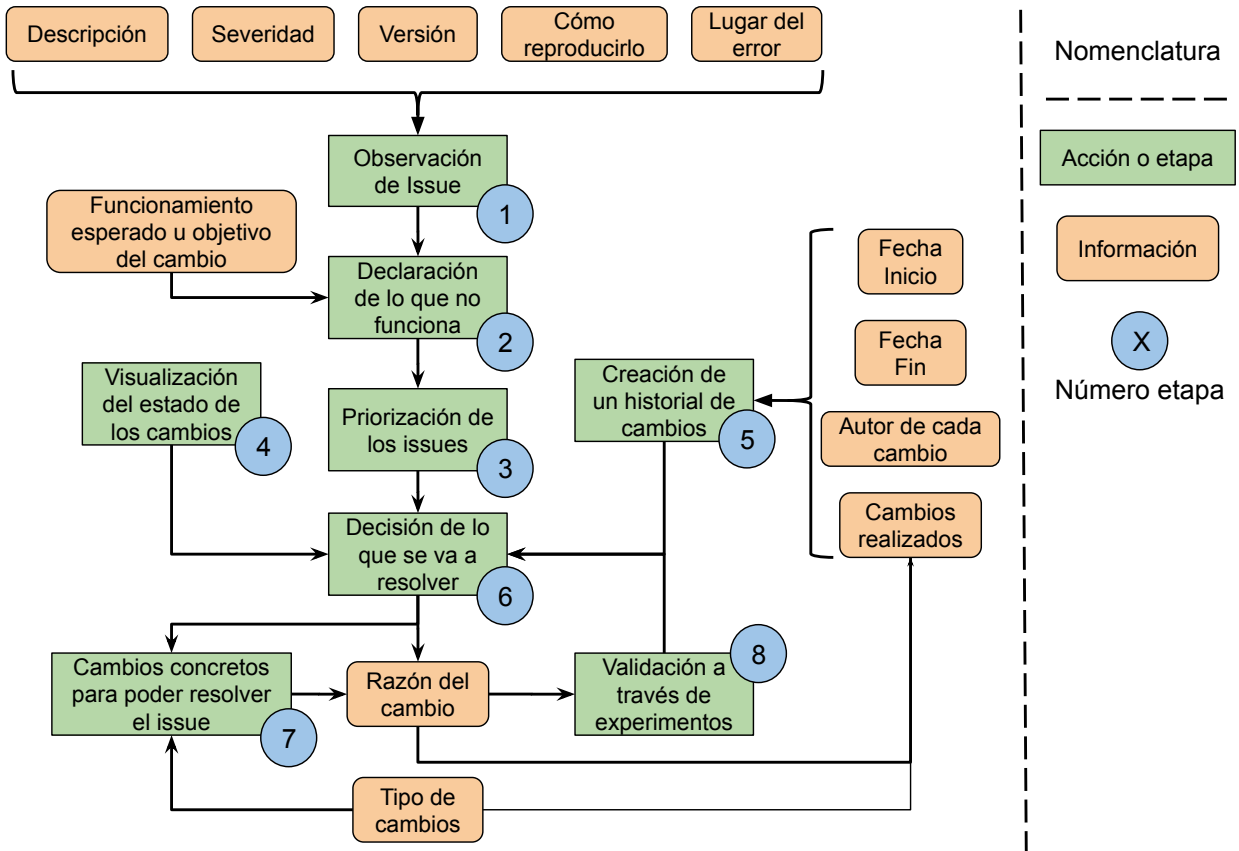


Figura 2.3: Diagrama resumen de las relaciones entre los distintos elementos que conforman un Sistema de Seguimiento de Incidentes o *Issue Tracker* en inglés.

demoró o se ha demorado.

- Quién hizo qué cambio, es decir qué cosas se han realizado y por qué usuario, en particular quién reportó, es decir que posee mayor información al respecto.
 - Qué fue lo que se modificó de manera concreta, es decir los cambios realizados.
6. Decidir qué incidente se va a resolver, en función tanto del orden, estado e historial de cambios.
 7. Definir concretamente los cambios o modificaciones que se van a realizar para poder resolver el incidente. En particular se identifica el tipo de cambio a realizar.
 8. Entender la razón detrás de un cambio. Esta decisión debiese ser validada a través de algún tipo de experimento que entregue mayor información al respecto, lo que podría modificar el paso 6.

Un *Issue Tracker* permite caracterizar los incidentes asociados al mal funcionamiento de un proyecto de software, el cual depende del código fuente que lo compone. A su vez, como se vio en la sección 1.4, el *CdC* se asimila al código fuente de los talleres realizados, por lo que es posible decir que existe una equivalencia entre un proyecto de software y la realización de dichos talleres. En otras palabras, la caracterización de incidentes aplicada a la realización de talleres permite identificar el mal funcionamiento de los talleres y por ende mejorarlo. Esto se relaciona directamente con la gestión del *CdC*, que es el problema a resolver en este documento.

2.4.2. Discovery and Delivery

Se refiere al flujo de trabajo propuesto por Optimizely [29], que considera el análisis y diseño o descubrimiento (en inglés *Discovery*), que luego se intersecta con la entrega o (*Delivery* en inglés). Es sumamente útil cuando el trabajo de análisis, diseño e implementación hace parte del mismo flujo.

Concretamente, la fase de *Discovery* tiene por objetivo entender y solucionar el problema al que se enfrenta un usuario. El producto final es un diseño terminado que responde a la necesidad del usuario. Luego de terminada la fase de diseño, se pasa a la fase de *Delivery*, la cual se encarga de implementar la solución diseñada.

Si bien este flujo no parece particularmente innovador, la gracia radica en que los procesos de investigación y diseño son parte explícita del flujo de trabajo, el cual involucra a todo el equipo, tanto aquellos que diseñan como aquellos que implementan.

La fase de descubrimiento se separa en dos etapas:

- Investigación o entendimiento del problema: se recopila información sobre el problema de manera de acotarlo.
- Diseño o exploración de soluciones: una vez acotado el problema, éste se plantea de manera explícita, acompañado de todos los casos de uso y el usuario objetivo. Todo aquello que queda fuera del alcance de la solución también debe ser considerado.

Dada la participación de todo el equipo, tanto incertidumbres técnicas como incertidumbres de acuerdo son resueltas durante la fase de descubrimiento. Esto permite que los encargados de implementar no tengan dudas al momento de hacerlo, lo que agiliza el proceso de desarrollo.

Así mismo como los desarrolladores estaban involucrados en el proceso de descubrimiento, los diseñadores también forman parte de la fase de entrega. Esto habilita a los diseñadores a asegurarse que la solución que se está implementando cumple con las características necesarias para responder a la necesidad del usuario.

A fin de cuentas, este proceso permite tener una mirada integrada de cómo la solución implementada responde al problema, desde la colaboración de un equipo multidisciplinario.

2.4.3. Trello

Trello es un software de administración de proyectos con interfaz web creado para gestionar las tareas que componen dicho proyecto [22]. La herramienta *Trello* permite hacer uso de la metodología de Tablero Kanban [10]. Esta consiste en una serie de estados que denotan un flujo, y una serie de tareas que pasan de un estado a otro. Esto es representado por distintas listas en forma de columnas (flujo), y tarjetas que pueden moverse de una columna a otra (tareas).

Cabe resaltar, que las columnas o estados posibles, poseen un límite sobre la cantidad de tarjetas que pueden poseer, también denominado “Trabajo a medio hacer” o *Work in Progress*, a partir de ahora *WiP*.

El tablero Kanban más simple corresponde a aquel que tiene 3 columnas: **To Do**, referente a todo el trabajo por realizar; **Doing**, es decir el trabajo que se está realizando; y **Done**, o el trabajo ya realizado.

La ventaja que tiene esta plataforma es la flexibilidad a la hora de definir nuevos estados de un flujo, al mismo tiempo que permite visibilizar el trabajo a todo aquél que participe del tablero. En este caso, al resto del equipo, tanto de desarrollo como de los colaboradores.

2.5. Resumen

Al finalizar este capítulo, el lector posee el trasfondo teórico para entender el resto del documento, en particular el detalle sobre herramientas de versionamiento que serán adaptadas al contexto de la empresa; las distintas herramientas de software utilizadas para desarrollar la solución; las metodologías que sustentan la totalidad de este trabajo; además de los flujos y maneras de modelarlos.

Capítulo 3

Determinación del Problema

Este capítulo da una visión en detalle del problema abordado. EN la sección 3.1, se presenta la situación actual, en particular los actores, objetos y procesos involucrados. Estos consisten en generar entendimiento en los aprendices, formar a los facilitadores y finalmente gestionar el cambio del *CdC*. Este último proceso es el que presenta la mayor cantidad de problemas, por lo que en la sección 3.2 se separa en distintos subproblemas, que abarcan desde la decisión de realizar una modificación, pasando por el seguimiento de estas, hasta poder cerrar el proceso y notificarlo al equipo. Este desglose permite caracterizar el comportamiento deseado por la solución, representado a través de criterios de aceptación. Acto seguido, se discute la relevancia de la solución en la sección 3.3, poniendo énfasis en los beneficios que conlleva mejorar el proceso de cambio, en particular al facilitar la creación de nuevo contenido. El capítulo termina con la sección 3.4 presentando los criterios de calidad, los cuales permiten luego poder priorizar las distintas alternativas de solución y/o implementación.

3.1. Situación Actual

Para entender la situación actual, es necesario entender qué actores interactúan con qué elemento u objeto, dentro de qué proceso están estos últimos insertos, de qué manera se interactúa con ellos y por qué razón u objetivo se realiza esto.

Para entender de mejor manera dicho contenido, se presentan a continuación las distintas entidades descritas en el párrafo anterior, para luego ver cómo se relacionan entre sí. De manera concreta, se presentan primero los actores, seguidos de los objetos con los cuales interactúan, para finalmente mostrar los procesos en los cuales se insertan.

3.1.1. Actores

Los actores son personas que interactúan con distintos objetos y de distintas maneras, dentro de una variedad de procesos. Estos son los siguientes:

1. Colaborador: se refiere a todos los integrantes de la compañía, independiente del rol que puedan tener en ese momento.
2. Facilitador: es aquel que facilita las sesiones/dicta los talleres. Su rol es el de guiar a los aprendices a lo largo de las distintas sesiones, facilitando la integración del nuevo conocimiento a través de distintos modos de aprendizaje.
3. Diseñador educacional: es aquel que tiene como rol diseñar el contenido educacional que se presentará durante las talleres. Nótese que el colaborador se transforma en diseñador educacional al realizar cambios o modificaciones en los contenidos educacionales.
4. Aprendiz: es aquel que participa de los talleres. Su objetivo es adquirir aprendizajes a partir de lo que le presente el facilitador durante las sesiones que atienda.

3.1.2. Objetos

Los objetos son todos aquellos elementos con los cuales los actores interactúan. Dado que la palabra “objeto” es sumamente general, se presentarán a continuación distintas categorías, la primera corresponde al tipo de objeto, y la segunda a la forma.

Tipos de Objeto

1. Objetos Contenedor: son aquellos que, valga la redundancia, actúan como contenedores de contenido.
2. Objetos Contenido: son aquellos que, de manera inversa a la categoría anterior, actúan como contenido perteneciente a contenedores.

Forma de los Objetos

1. Objetos Explícitos: son aquellos que están totalmente definidos y que existen, ya sea de manera física o virtual. Un ejemplo de esto serían las presentaciones.
2. Objetos Implícitos: son aquellos que no están del todo, o nada definidos. En otras palabras, aún no toman forma, y al no poseer características definidas es difícil trabajar con ellos. Parte del trabajo a realizar durante esta memoria es justamente definir dichos objetos, para que pasen de implícitos a explícitos.

Ahora que las distintas categorías de objetos han sido presentadas, se explicarán los distintos objetos presentes, acompañados tanto de su tipo como de su forma:

Lista de objetos

1. Taller: se trata de un contenedor explícito que contiene una cierta cantidad de módulos. El taller es diseñado específicamente para responder a las necesidades de un cliente de LeanSight.

2. Módulo: nuevamente se trata de un contenedor explícito, el cual contiene una cantidad específica de aprendizajes. La idea detrás de un módulo es poder conferir conocimientos específicos a los aprendices.
3. Sesión: contenido explícito correspondiente a una instancia temporal durante la cual un facilitador realiza uno o más módulos a un público compuesto de aprendices. Una sesión tiene asociado una fecha de realización, un cliente y un conjunto de módulos.
4. Aprendizajes: contenedor implícito de una variedad de elementos que entregan aprendizajes. Estos elementos pueden ser contenido educativo, medios o experimentos, pero dado el carácter implícito, por ende indefinido, de este objeto, puede que contenga otros elementos.
5. Presentación: contenido explícito y soporte visual de los aprendizajes durante las capacitaciones. El facilitador se basa en la presentación para ayudar a construir el conocimiento de los aprendices. Parte importante de esta construcción está ligada al cuadernillo. La presentación está implementada en *Slides*.
6. Cuadernillo: contenido explícito complementario a la presentación. Éste define los objetivos de aprendizaje generales y específicos del módulo al que pertenece, además de permitir ya sea reflexionar o poner en práctica estos últimos. Es también una herramienta que permite repasar los contenidos a posteriori, dado que cada aprendiz posee su propio cuadernillo. Éste está implementado en MS Word.

A modo de resumen de lo anterior, se introduce la figura 3.1, la cual presenta las relaciones entre los distintos actores, y los objetos presentados hasta ahora.

1. Contenido Educativo: contenido explícito inserto en la presentación y/o cuadernillo, además de estar contenido dentro de los aprendizajes. Generalmente se traducen en principios de solución.
2. Medios: contenido explícito inserto en la presentación y el cuadernillo, además de estar contenido dentro de los aprendizajes. Estos consisten en documentos, videos o enlaces, que tienen distintos propósitos:
 - Dar a entender una situación particular que habilite realizar una distinción propia del módulo.
 - Entregar un argumento de autoridad a partir de fuentes externas.
3. Experimentos: contenido explícito inserto en la presentación y/o cuadernillo, además de estar contenido dentro de los aprendizajes. La idea de un experimento es comprobar la existencia de un problema, dentro de un ambiente controlado o simplificado, para luego poder introducir un principio de solución que permita resolverlo.
4. Guía del Facilitador: contenido explícito que busca entregar una estructura base del relato, o lógica a seguir, durante la capacitación. Además, es aquí dónde se muestra de manera explícita la relación entre la presentación, el cuadernillo y/o los aprendizajes, haciendo énfasis en el objetivo detrás de cada una de estas relaciones. Cabe resaltar que la Guía del Facilitador es una herramienta que está siendo probada, dado que existe una sola instancia de ella, asociada a un módulo en específico.
5. Checklist: contenido explícito perteneciente a un módulo. El objetivo de este elemento es listar los materiales que son necesarios para llevar a cabo dicho módulo. Ejemplo de esto son sharpies, post-its, el cuadernillo, etc.

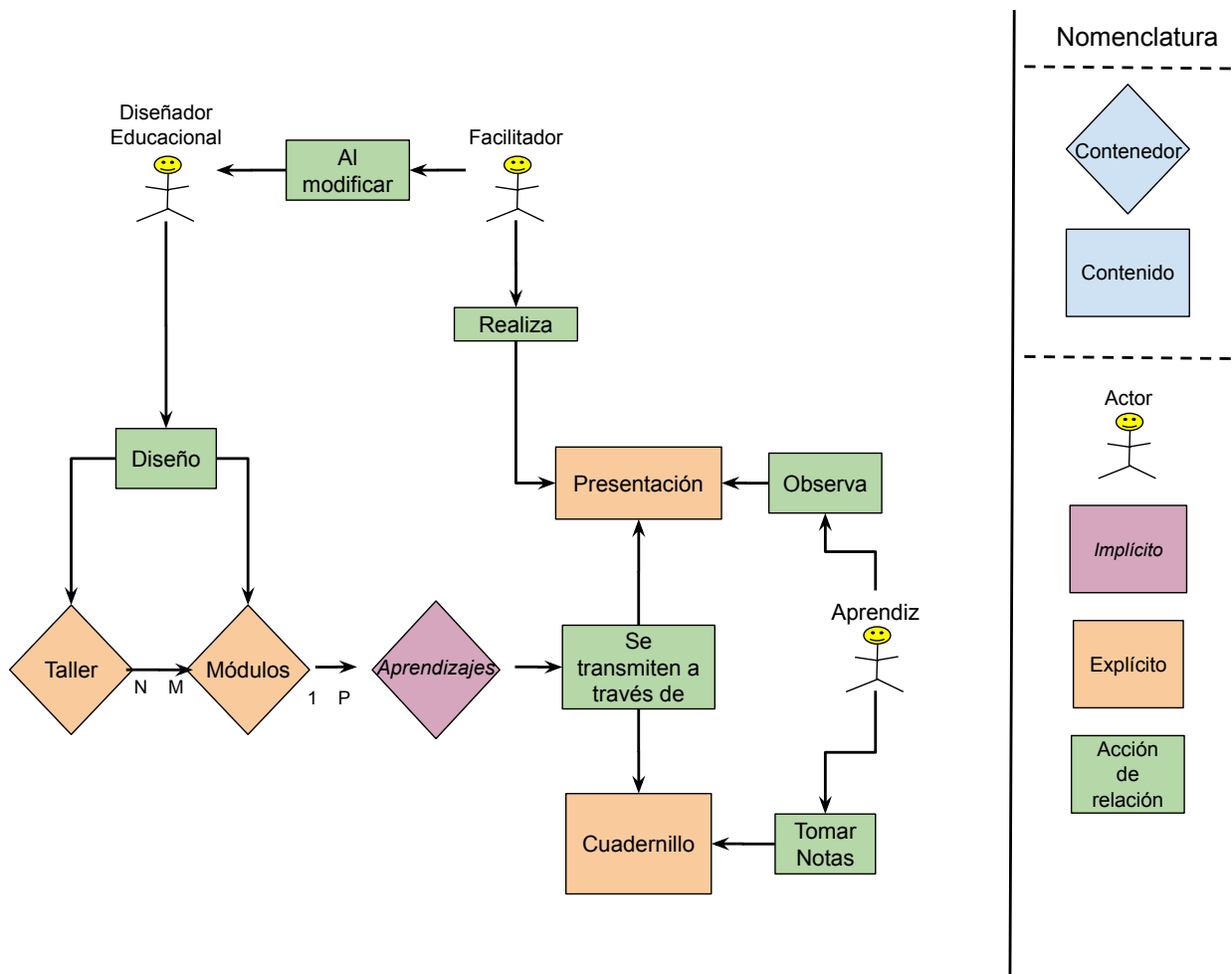


Figura 3.1: Diagrama simplificado de las interacciones entre los distintos actores.

6. Cambios: contenido implícito presente en todos los elementos anteriores. El objetivo del cambio es mejorar el contenido. Hay distintos tipos de cambio:

- Cambio de un módulo desde un taller a otro.
- Agregar o quitar un aprendizaje de un módulo.
- Modificar un aprendizaje para que se entienda mejor.

Nótese que el cambio, al ser implícito, no está del todo definido, por lo que hay muchos tipos de cambio que no están denotados aquí. Uno de los objetivos de esta memoria es justamente definirlos y poder transformar este elemento en un objeto explícito.

3.1.3. Procesos

Si bien el lector ahora conoce el rol y objetivo de cada uno de los objetos y actores, un diagrama general que resuma las distintas interacciones entre estos últimos puede resultar complejo de entender. Es por ello que a continuación se presenta un diagrama simplificado de las relaciones entre los distintos procesos, para luego indagar en mayor profundidad en cada uno de ellos.

Los procesos involucrados son los siguientes:

1. Generando entendimiento: tiene como objetivo que el aprendiz adquiera conocimiento. Depende tanto del facilitador como del contenido que este tiene a su disposición.
2. Formación de facilitadores: tiene como objetivo formar a los facilitadores “Junior” para que puedan facilitar la adquisición de los aprendizajes de mejor manera. Depende tanto de la experiencia del facilitador “Senior” que lo apoye como del contenido que tiene a su disposición.
3. Diseño y rediseño (o cambio): tiene por objetivo mejorar la calidad del contenido presentado por el facilitador, lo cual a su vez busca mejorar la adquisición de conocimientos de los aprendices. Este proceso de mejora incide directamente tanto en la generación de entendimiento como en la formación de los facilitadores, quienes disponen de mejores herramientas en forma de contenido y/o dinámicas de aprendizaje que aplicar.

En la figura 3.2 se presenta el funcionamiento a grandes rasgos y la interdependencias de los procesos recién descritos, además de su relación con el propósito de la empresa. Cada proceso se explicará con mayor detalle en las siguientes subsubsecciones.

En caso de querer indagar más en detalle en las interacciones entre los distintos elementos presentados anteriormente, se deja en anexo B el diagrama general detallado de la situación actual.

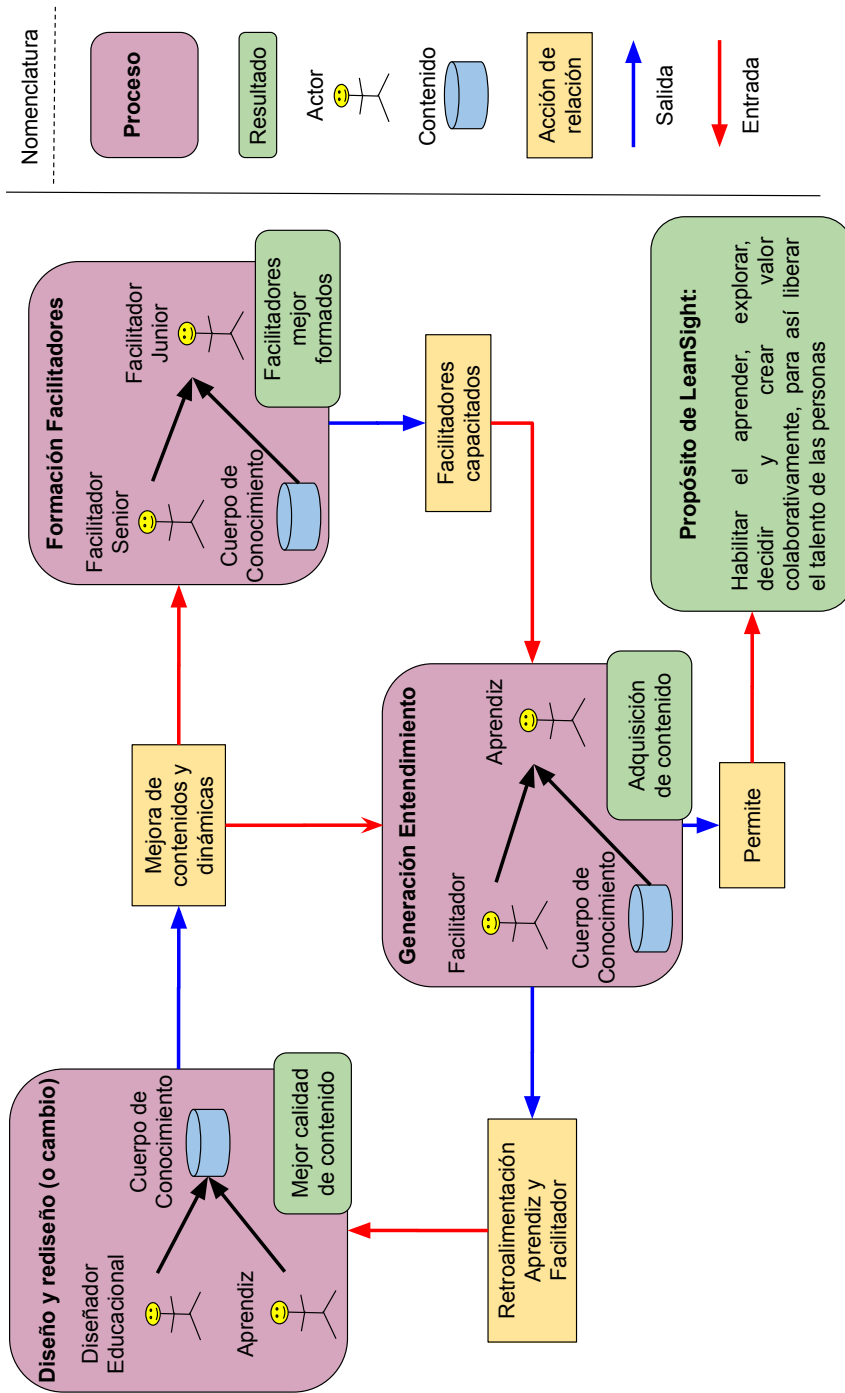


Figura 3.2: Diagrama simplificado de la relación entre los distintos procesos dentro de *LeanSight*, v20181211.

Generando entendimiento

Tal y como se plantea en el capítulo 1, el objetivo de la compañía es mejorar la cultura empresarial a través de talleres que entreguen una serie de competencias ⁵. Para lograr esto, se busca generar entendimiento, desde el cual los aprendices puedan apropiarse del contenido. Esto se realiza en base a un experimento, el cual plantea una situación hipotética que es resuelta a partir de un principio de solución.

No obstante, es difícil determinar si estos experimentos generan realmente entendimiento. Es por esto que se hace hincapié durante las sesiones en que los aprendices escriban y tomen nota en sus cuadernillos de lo que vayan aprendiendo. Esta y otras observaciones se consideran como alertas.

Resumiendo lo anterior, se presenta el proceso de generación de entendimiento en la figura 3.3. Nótese que se agregaron alertas, las cuales denotan un problema dentro de un proceso o interacción entre distintos objetos; además de objetivos, que permiten entender el rol del objeto.

Formación de facilitadores

Con el objetivo de poder capacitar nuevos integrantes en la realización de sesiones, es que se creó la Guía del Facilitador, la cual busca entregar todos los conocimientos necesarios para poder facilitar un módulo. Esto incluye un checklist, con todos los materiales necesarios para llevar a cabo la sesión. Luego de ser entregada esta información, el nuevo integrante debiese poner en práctica el conocimiento adquirido. Acto seguido, el proceso de aprendizaje y realización de sesiones es repetido, lo que permite afinar el hilo conductor, o relato, con el que el nuevo facilitador realiza las sesiones.

Este proceso, representado en la figura 3.4 incluye dentro de sí el proceso de generación de entendimiento planteado previamente, al cual se agrega el rol de formación de la guía del facilitador, además del rol verificador del checklist.

Diseño y rediseño (o cambio)

Los talleres son diseñados y rediseñados constantemente por los diseñadores educacionales. El proceso de diseño implica tomar decisiones sobre qué módulos pertenecen a qué taller, y consecuentemente qué aprendizajes harán parte de qué módulo. El proceso de rediseño se sustenta en la búsqueda por parte de la compañía de mejorar la manera de generar entendimiento en los aprendices, y se traduce en forma de cambios.

No obstante, hay varias alertas que es necesario señalar, tales como la falta de definición de lo que es aprendizaje, y de como este puede migrar de un módulo a otro. Otras alertas sumamente importantes se dan al realizar modificaciones, ¿cómo es que el resto del equipo se

⁵Estas son el aprender, explorar, decidir y crear valor colaborativamente.

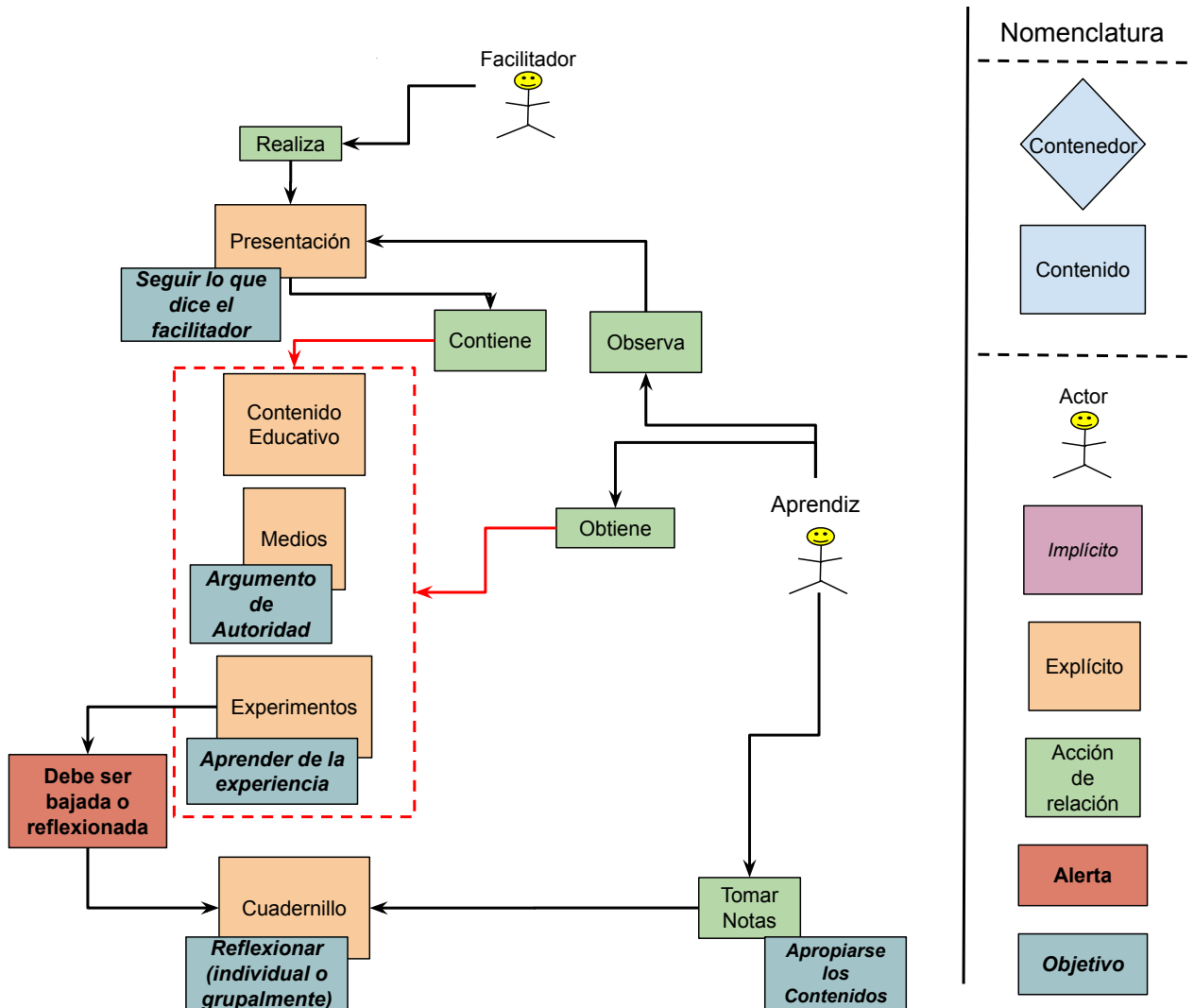


Figura 3.3: Diagrama explicativo del proceso que busca generar entendimiento en los aprendices v20181211. El recuadro rojo indica una agrupación entre el contenido educativo, los medios y los experimentos.

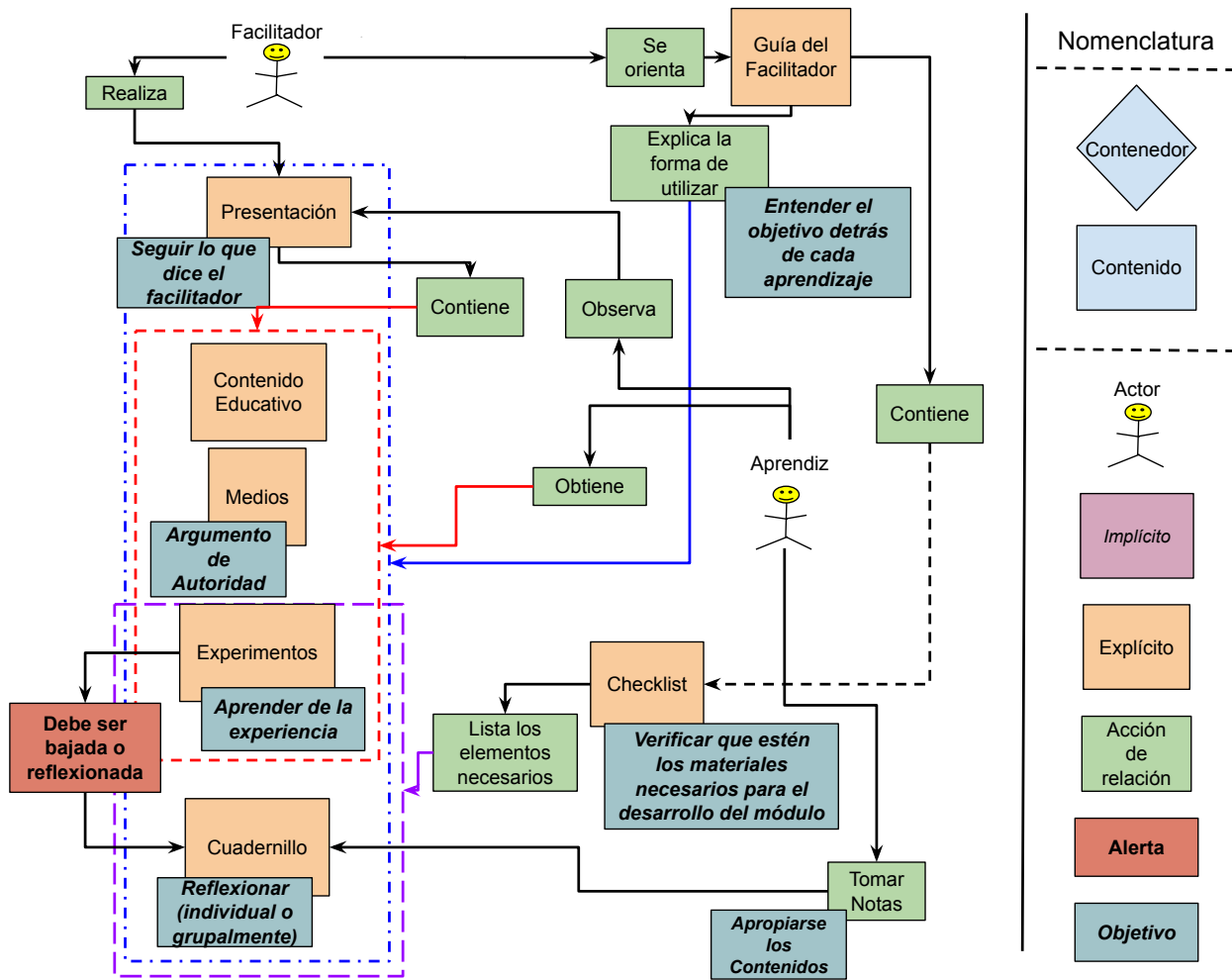


Figura 3.4: Diagrama explicativo del proceso que busca formar facilitadores v20181211. Los recuadros de colores indican distintas agrupaciones. En lo que respecta a la flecha punteada, actúa de la misma manera que las flechas negras, pero sirve para diferenciar ambas flechas en caso de intersección.

entera de que se realizan modificaciones? ¿Por qué razón se hacen dichas modificaciones? ¿En qué se sustentan? Y más específicamente, ¿cómo es posible gestionar el proceso de rediseño o modificación de los elementos del *CdC*, es decir el proceso de cambio?

En la figura 3.5 es posible ver en forma de diagrama el proceso de cambio.

3.1.4. Acotando el problema

Como es posible notar en el diagrama general de la situación actual (figura B.1), de los tres procesos planteados en la sección 3.1.3, el proceso de cambio es aquél que presenta la mayor cantidad de alertas y de objetos intangibles, en otras palabras, es el proceso menos comprendido.

Por otro lado, los procesos de formación de facilitadores y generación de entendimiento, presentan una componente pedagógica fuerte, la cual escapa del dominio de las ciencias de la computación. Si bien estos procesos podrían beneficiarse de aportes metodológicos, se decide no indagar en ellos.

En resumen, el proceso de cambio, también conocido como el proceso de diseño y rediseño, es aquél estudiado a continuación.

3.2. Delimitando los problemas

3.2.1. ¿Cómo es posible conocer la situación actual de un módulo?

Los módulos consisten en una serie de elementos relacionados, en particular la presentación y el cuadernillo conforman gran parte del contenido a presentar. No obstante, estos recursos son extensos: una presentación posee entre 40 a 60 diapositivas, un cuadernillo entre 10 a 20 páginas, lo que dificulta poder entender la lógica detrás de ellos.

Ahora bien, el propósito de mejorar los contenidos se plasma a través de modificaciones a los contenidos, por ende a la lógica. Si se define el cambio, como los pasos a seguir para pasar de una situación actual a una situación objetivo, entonces para poder realizar cambios, se hace necesario identificar tanto la situación actual como la situación objetivo. ¿Pero, cómo es posible identificar dichas situaciones?

La respuesta que se presenta a continuación contempla el hecho de visibilizar la situación actual de la lógica de un módulo, también conocido como relato. La visibilización permite, en primera instancia, plasmar lo que un colaborador entiende, para luego compartir dicha información con el resto del equipo. Otros colaboradores pueden discutir o modificar la información presentada, hasta llegar a un consenso. A partir de este consenso, es posible definir la situación actual, y/o la situación objetivo, para finalmente definir concretamente los pasos que se van a tomar para pasar de una a la otra. Cabe resaltar, que si un colaborador no está

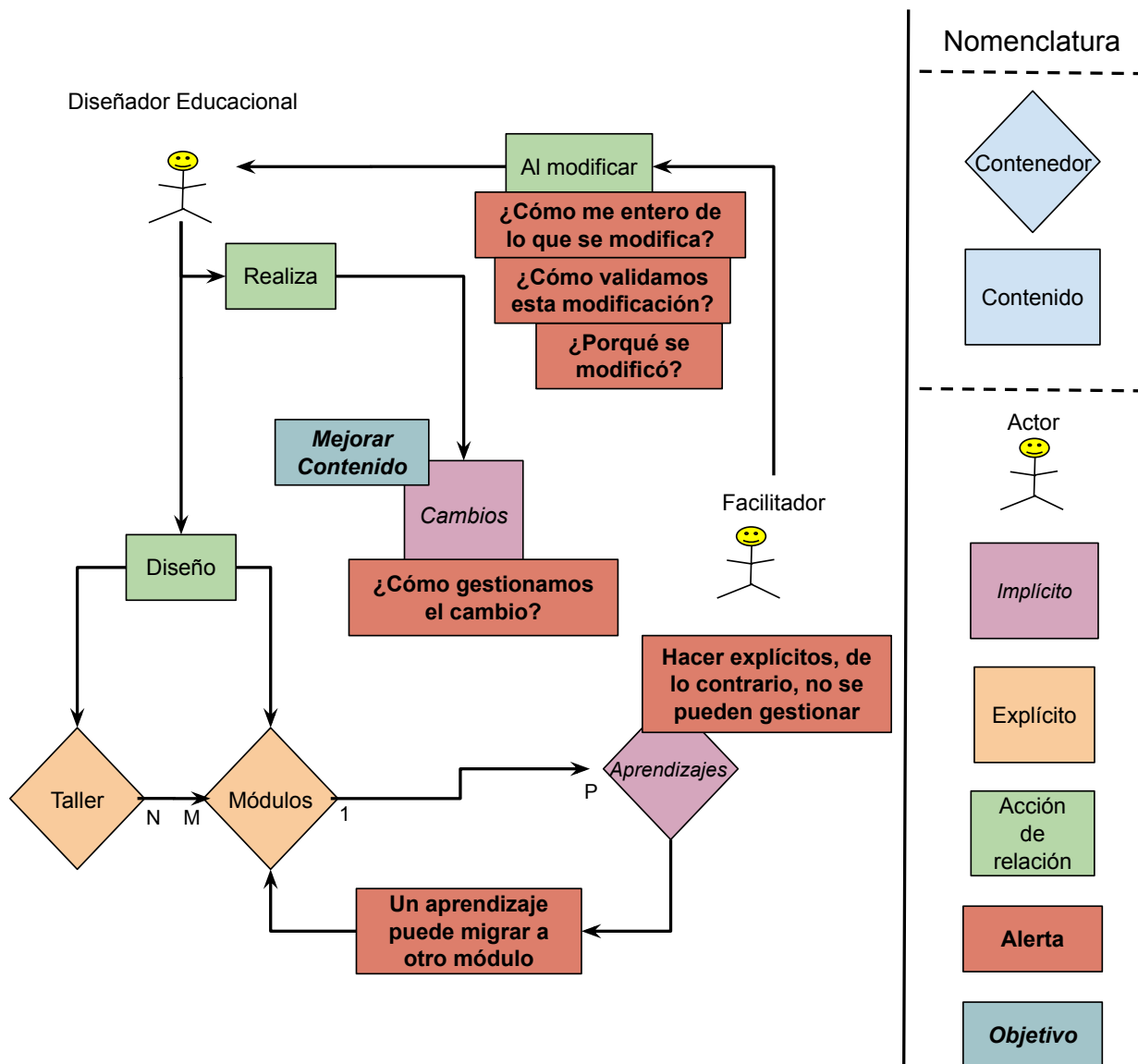


Figura 3.5: Diagrama explicativo del proceso de diseño y rediseño del *CdC*, también conocido como el proceso del cambio v20181211.

presente durante la discusión, la visualización del relato le permitiría entender la estructura del módulo, por lo que también sirve como herramienta de comunicación al interior del equipo.

De manera concreta, los pasos a seguir se traducen en modificaciones a los contenidos, o al orden de estos. En particular, poder navegar el orden y ubicación de los distintos contenidos permite identificar los elementos existentes, además de las relaciones entre los distintos contenidos. Otra característica importante, es la capacidad de gestionar y priorizar los cambios por realizar a los módulos. Esto es posible a partir de la visión general de los contenidos y los cambios que se quieren realizar, es decir a partir de la visibilización del relato, el cual define la situación del módulo.

De lo anterior se desprenden los siguientes criterios de validación, presentes en el anexo A: F-1, DE-1, DE-2, DE-3, DE-4.

3.2.2. ¿Qué información es necesaria para realizar modificaciones que permitan mejorar el contenido?

Al trabajar de manera individual, dicho individuo posee la totalidad de la información, por lo que es capaz de tomar decisiones acertadas. En el caso de un grupo de personas, la información esta fragmentada entre los distintos individuos. Es por ello que nace la necesidad de poner en común la información recopilada.

Esto permite no solo tomar decisiones de manera individual, basándose en la información del resto, si no que también podría habilitar la paralelización del trabajo, si se da que hay suficiente información compartida y existe confianza en el criterio del resto de los colaboradores.

En este caso, dicha información corresponde a inicialmente una observación, la cual está inserta dentro de un contexto, lo cual caracteriza un problema. Si no se comunica de buena manera el problema, es muy difícil llegar a una solución que lo resuelva, por lo que es fundamental compartir la información correcta. Sin embargo, ¿cuál es esta información?

Por otro lado, si se conoce el problema, hay que saber situar de manera concreta los cambios que representa, además de evitar modificaciones concurrentes, o volver a realizar ediciones que ya tuvieron lugar, sin olvidar verificar la correctitud de los cambios. Esta información debe ser accesible de manera ordenada por todo el equipo, lo que habilita organizarse para llevar a cabo los cambios pertinentes, en el orden más adecuado.

De lo anterior se desprenden los siguientes criterios de validación, presentes en el anexo A: F-2 , DE-5 , DE-6 , DE-7 , DE-8 , DE-9 , DE-10 , DE-11 , DE-12 y DE-13.

3.2.3. ¿De qué manera es posible comprobar el estado y propagación del cambio de los contenidos?

Si bien pueden existir diversas estrategias para comprobar el estado del cambio, la más sencilla consiste en poder visualizar la totalidad del contenido en su formato de edición. De manera concreta, esto significa poder ver todos los contenidos al mismo tiempo, para así evaluar qué se debe cambiar, dónde se realiza dicha modificación y cuán importante es.

Por otro lado, al visualizar los contenidos, es directo saber si un cambio ya fue realizado o no, además de poder verificar si un cambio se propaga a otro soporte o no. De manera global, esto permite verificar que las modificaciones estén bien hechas.

No obstante, dado que hay por lo menos dos soportes, relacionados entre sí, que editar al mismo tiempo, y que estos son extensos ⁶, el hecho de visibilizar los recursos mientras se modifican es un problema complejo.

De lo anterior se desprenden los siguientes criterios de validación, presentes en el anexo A: DE-14 , DE-15 y DE-16.

3.2.4. ¿De qué manera se puede mejorar el proceso de publicación de los recursos de una sesión o módulo?

El subproceso de publicación consiste actualmente en transformar los soportes desde un formato de edición a un formato apropiado para realizar sesiones. Si bien esto parece sencillo, se trata de una serie de pasos engorrosos:

1. Publicar la presentación en formato .pdf y .pptx.
2. Asociar los medios o recursos de manera local al .pptx de la presentación.
3. Publicar el cuadernillo en formato .pdf.
4. Imprimir una copia del cuadernillo, por cada módulo, por aprendiz.
5. Insertar el(los) cuadernillo(s) impreso(s) dentro de una carpeta, para cada aprendiz.
En caso de haber más de un módulo, es necesario considerar también el orden en que estos serán dictados, y por ende poner los módulos en orden dentro de la carpeta.

Lo anterior habilita el poder realizar la sesión sin acceso a internet y que cada aprendiz posea su cuadernillo propio desde el cual poder tomar nota y realizar las actividades.

Actualmente, con el fin de asegurar la existencia de una instancia actualizada y publicada, es que se está constantemente publicando toda instancia de sesión o módulo que se crea. Este proceso toma tiempo, por lo que idealmente solo debiese ser realizado en caso de necesidad. De aquí nace la pregunta ¿cuándo es necesario publicar una instancia de sesión o módulo?

A modo de resumen, es necesario disminuir tanto el tiempo como la cantidad de pasos necesarios para publicar una instancia de sesión en el formato adecuado para ser presentada.

⁶Un cuadernillo posee entre 10 a 20 páginas, una presentación tiene entre 40 a 60 diapositivas.

Esto incluye que la sesión pueda ser realizada sin acceso a internet, es decir de manera local.

De lo anterior se desprenden los siguientes criterios de validación, presentes en el anexo A: F-3 , F-4 , F-5 y F-6.

3.2.5. ¿Cómo es posible saber qué cambió luego de una modificación a un contenido?

Con el propósito de mejorar el contenido, este es modificado de manera recurrente. No obstante, es difícil que todo el equipo sepa si existe relación entre las distintas modificaciones. Si bien un cambio aislado no necesariamente denota una intención clara, un conjunto de cambios específicos y relacionados podría aportar mayor información. Es por ello que ser capaz de diferenciar entre conjuntos coherentes de cambios podría ayudar a entender las razones que los justifican.

Por otro lado, si existen conjuntos coherentes de cambios, al realizar todas las modificaciones asociadas, es posible declarar estabilidad en los soportes, además de identificar los cambios concretos que se realizaron dentro de dicho conjunto.

Esta información es sumamente útil para saber de qué manera cambió el soporte, por qué razón se realizó el cambio además de dar la posibilidad de publicar dicha instancia estable en caso de necesidad.

De lo anterior se desprenden los siguientes criterios de validación, presentes en el anexo A: DE-17 , DE-18 , DE-19 y DE-20.

3.2.6. ¿De qué manera es posible diferenciar entre distintas instancias de un contenido?

Actualmente, se realiza un gran volumen de sesiones ⁷ a públicos distintos, público al que se envían los soportes de la sesión a la cual asistió.

No obstante, cabe la posibilidad de que el contenido de dichos talleres varíe entre realizaciones. En otras palabras, cada instancia de taller posee un contenido y un público específico, esto corresponde a una sesión.

Con el objetivo de poder compartir una sesión específica al público asociado o simplemente poder comparar la efectividad de los contenidos, es que se hace necesario poder acceder de manera ordenada a las distintas sesiones.

De lo anterior se desprenden los siguientes criterios de validación, presentes en el anexo A: F-7 , F-8 , F-9 , F-10 y F-11.

⁷Al realizar una rápida inspección del calendario de la empresa, se estima que en promedio se realizan entre 4 a 5 sesiones de media jornada semanales.

3.2.7. ¿Cómo es posible mejorar el proceso de cambio?

El propósito detrás del proceso de cambio es el de mejorar el contenido educativo de la empresa, el cual se encuentra principalmente en dos formatos: las sesiones y los módulos. En otras palabras, el propósito del proceso de cambio se podría enunciar como el diseño y construcción de módulos y sesiones.

Así mismo, en el caso en que se puedan solucionar los problemas planteados en esta sección, es posible hablar de una simplificación del diseño, por ende de facilitación y/o mejora del proceso de cambio. A continuación se presenta la lógica general.

El hecho de visibilizar la lógica o relato de un módulo, permite definir la situación de éste. Esto es válido tanto para situaciones actuales como objetivo, por lo que se habilita la capacidad de comparar el estado de un módulo con otro. Esto se plantea en el apartado 3.2.1.

Con la capacidad de definir la situación o estado de un módulo, es posible caracterizar la situación objetivo. No obstante, esta debe ser justificada a partir de observaciones de comportamientos no deseados del contenido. Es por ello que se hace necesario saber qué tipo de información es pertinente entregar. Esto habilita entender el porqué detrás de cada modificación. Esto se define en la sección 3.2.2.

Ahora que se manejan los estados y las razones que justifican el pasar de un estado a otro, es posible modificar los contenidos de forma consciente. No obstante, dada la interdependencia entre los distintos soportes de un mismo módulo se hace necesario poder visibilizar la totalidad de los recursos. De esta manera, no sólo se puede verificar que las modificaciones sean correctas, sino que se evidencia la propagación del cambio a los distintos soportes. Esto se describe en la sección 3.2.3.

Ahora es posible identificar el estado inicial, las razones para pasar al estado objetivo son visibles y existe la capacidad de realizar cambios considerando la propagación de estos. No obstante, para realizar una sesión, es necesario tener los recursos en un formato adaptado, por lo que es necesario publicarlos. El proceso de publicación es por ende clave para tener un producto utilizable. De simplificarse el proceso y/o acortarse la cantidad de pasos, entonces es posible reducir los tiempos de publicación y por ende dejar más tiempo disponible para otras actividades de los colaboradores dentro de la empresa. Esto está considerado en la sección 3.2.4.

Al dar solución a los problemas planteados previamente, ya se tiene una mejora del proceso de cambio. No obstante, si bien es posible conocer las razones para pasar de un estado a otro, no es posible aún conocer qué cambios fueron realizados en una instancia en particular de un módulo. Es por ello que se hace necesaria la capacidad de agrupar cambios coherentes, para acompañarlos de un resumen de los cambios notables. Esto se plantea en la sección 3.2.5.

Finalmente, es necesario poder distinguir entre instancias de sesiones y/o módulos, dado que cada instancia estará asociada a un estado o situación distinta. El hecho de diferenciar habilita el poder atribuir propiedades a una u otra instancia de los módulos o sesiones. Esto se presenta en la sección 3.2.6

Como es posible ver, al dar solución a todos los problemas presentados, se obtiene un mejor proceso de diseño y rediseño, también conocido como proceso de cambio.

De lo anterior se desprenden los siguientes criterios de validación, presentes en el anexo A: DEF-1 , DE-21 , DE-22 , DE-23 , DE-24 , DEF-2 , DE-25 , DE-26 y DE-27.

3.3. Discusión sobre la relevancia de la solución

Para entender la relevancia de la solución, es necesario recordar el propósito de *LeanSight*: *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*. Concretamente, esto se traduce en sesiones o talleres realizados a las empresas cliente. Pero estos talleres no siempre logran su objetivo, es por ello que se busca mejorar continuamente el contenido dictado, y esto a su vez se traduce en cambios al *CdC*.

Estos cambios, no sólo traducen una voluntad de la empresa de mantenerse actualizada, y por ende competitiva, sino que también pertenecen a la propuesta de valor de la compañía, al permitirle mantenerse en el estado del arte. Es por ello que es sumamente relevante, en una primera instancia, comprender y luego reformular el proceso de cambio del *CdC*, de tal manera de poder gestionar los cambios de manera simple, controlada, eficiente en tiempo y en trabajo. A la larga, esto habilitaría la capacidad de mejorar la calidad de los talleres, y por ende de los servicios, lo que significa una mejor imagen de la empresa.

Cabe resaltar que este trabajo procura abordar un problema real de una empresa, buscando entregar valor lo más temprano posible a esta última. Esto hace necesario priorizar la simplicidad de la solución, de tal manera de poder ser utilizada por todos los colaboradores. Es por ello que el foco no está en desarrollar código, sino más bien en el diseño, teniendo como objetivo proveer una buena experiencia usuario.

3.4. Características de calidad deseadas y/o requeridas por la solución

Esta sección aborda las características de calidad deseadas y/o requeridas de la solución. El objetivo de estas características es guiar las decisiones de diseño e implementación, son descritas a continuación.

3.4.1. Calidad de software

La calidad de software engloba todo lo referente a la calidad del software o código implementado. En otras palabras, características como escalabilidad, bajo acoplamiento y alta

cohesión, código testeado, responsabilidades claras al utilizar programación orientada a objetos, utilizar patrones de diseño, ser eficiente en tiempo y/o espacio, realizar el mínimo de llamadas a APIs externas, etc.

El objetivo detrás de esta característica es poder tener un código fácilmente mantenible, que pueda ser potencialmente mantenido por otros desarrolladores, además de permitir escalabilidad de la implementación.

La manera en que se verifica la calidad de código, es utilizando GitHub y GitFlow, acompañado de Revisión de Pares para cada *Pull Request*, antes de poder integrar las distintas ramas del código implementado a la rama principal. Por otro lado, se utiliza herramientas de chequeo de estilo como *TSLint* [15], además de documentar el código, y refactorizarlo periódicamente.

3.4.2. Experiencia usuario

La experiencia usuario o User Experience [1] (*UX*) se refiere a cómo se siente un usuario al interactuar con un sistema. En este caso, considera la experiencia de los colaboradores de *LeanSight* al participar del proceso de rediseño. Dentro de la *UX* se consideran las características descritas a continuación:

Abordabilidad por todo usuario

Contempla la capacidad de una solución de poder ser abordada o utilizada por cualquier usuario, independiente de su formación profesional. Esta característica es sumamente relevante a la hora de diseñar una solución, dada la variedad de formaciones de los integrantes del equipo, en particular aquellos que no vienen del área de la informática, los cuales representan cerca de un tercio del total de colaboradores.

Para comprobar la abordabilidad por todo usuario, se realizan validaciones intermedias en donde se integran todos los miembros del equipo, en particular aquellos usuarios que no tienen un trasfondo profesional ligado al desarrollo de software. Su retroalimentación se vuelve sumamente valiosa a la hora de diseñar la experiencia usuario.

Simplicidad de la solución

Se dice que una solución es simple cuando es fácil de entender y presente pasos concretos y sencillos. Si la solución no fuese simple, existiría la posibilidad de que los usuarios no la utilicen: ¿para qué agregar aún más complejidad y/o carga a su trabajo? La solución debe ser simple.

La simplicidad de la solución es dada por diseño, dada la metodología que aplica ciclos cortos de mejora continua, se busca siempre generar la solución más pequeña y simple que

entregue valor a los usuarios. Por otro lado, con las validaciones intermedias, y la validación final, es posible verificar cuán usable es la solución implementada, para luego ajustarla.

Utilizabilidad y testeabilidad de la solución

Una solución debe poder ser utilizada por los usuarios. Esto permite que dicha solución sea testeada, lo que a su vez podría generar retroalimentación sobre el funcionamiento de la solución. Esta información luego es utilizada para mejorar la solución y repetir el proceso nuevamente.

Esto es sumamente importante dentro del esquema de solución iterativa que se utiliza, donde la retroalimentación del usuario permite mejorar la solución y el entendimiento del problema.

La manera de comprobar la utilizabilidad y testeabilidad de la solución es a través de los ciclos cortos de desarrollo, donde cada ciclo posee una validación con usuarios de tal manera de ajustar la solución implementada y adaptarla a las necesidades de los usuarios y de la empresa. Por otro lado, esta característica se une con la característica de calidad de adaptabilidad de la solución, dado que el tiempo de los usuarios es limitado, por lo que soluciones que se adapten al flujo de trabajo actual les permite ponerlas a prueba en escenarios más parecidos a los reales. En otras palabras, esta característica influye dentro del diseño e implementación de la solución propuesta.

3.4.3. Diseño iterativo de la solución

La solución debe poder diseñarse de manera iterativa. Para poder cumplir con esta características, a continuación se presentan una serie de requisitos.

Producto semilla

El producto semilla es aquél producto que soluciona el problema de la manera más sencilla, pero que además posee las características necesarias para escalar a la solución final. Es similar al concepto de Producto Mínimo Viable (*PMV* o *MVP* por sus siglas en inglés⁸), sumando el hecho de poseer las características para escalar.

La manera de comprobar esta característica de calidad es a través del diseño de la solución, a través de ciclos cortos de desarrollo, donde cada ciclo tiene por objetivo incrementar la solución actual y adaptarla tanto a los usuarios como a las necesidades de la empresa.

⁸Minimum Viable Product

Incrementabilidad de la solución

El concepto de solución incremental está ligado a aquél de producto semilla dentro del marco del desarrollo iterativo. Un producto semilla posee todas las características para escalar a la solución final, pero va “creciendo” por incrementos.

Esta característica de calidad es fundamental para asegurarse que el producto semilla vaya integrando cada vez más funcionalidades, por lo que determina cómo se realiza el diseño e implementación de la solución.

Entrega temprana de valor

Denota la capacidad de la solución de entregar valor a la empresa desde etapas tempranas de su implementación. En otras palabras, luego de cada ciclo de diseño, y en función del entendimiento en ese momento del problema, se prioriza implementar el incremento de solución que permita entregar el mayor valor a la empresa. En este caso, el valor se entiende como aprendizaje o entendimiento del problema, y a su vez como un aporte al proceso de cambio.

De la misma manera que los apartados anteriores, esta característica permite priorizar entre las funcionalidades por desarrollar, dónde aquella funcionalidad que requiera el menor trabajo y entregue el mayor valor es priorizada por sobre otras. En otras palabras, permite determinar el diseño e implementación de la solución.

Adaptabilidad al proceso actual

La solución implementada debe poder adaptarse al proceso actual, esto con el objetivo de no agregar demasiada carga de trabajo a los colaboradores y aún así poder ser utilizada y por ende validada. El ideal es poder realizar cambios a pequeña escala para ir migrando del estado actual al estado objetivo.

Para poder medir esta característica de calidad, es necesario realizar las validaciones intermedias presentes dentro del ciclo de diseño e implementación del ciclo de mejora. Cada validación intermedia permite determinar cuán distante está la solución o implementación de los procesos actuales de la empresa, para luego poder adaptarla de mejor manera a las necesidades de la empresa y por ende al flujo de trabajo de los colaboradores.

3.5. Resumen

Al finalizar este capítulo, se tiene conocimiento detallado del funcionamiento del proceso de diseño y rediseño, de los subproblemas que posee. A partir de éstos, es posible definir el comportamiento deseado de la solución a través de criterios de aceptación. En lo que respecta

las características de calidad, estas guían las decisiones de diseño e implementación. Todo lo anterior entrega un marco en el cual desarrollar este trabajo.

Capítulo 4

Descripción de la solución

En este capítulo se aborda la solución a cada uno de los subproblemas planteados en el capítulo anterior, utilizando la metodología de *Nemawashi*. En la sección 4.1, se describe la manera de diferenciar distintas instancias de módulos, al implementar versiones, lo que habilita el poder realizar distinguir entre elementos. En la sección 4.2, se determina la información necesaria para poder realizar los cambios, basada en el funcionamiento de un *Issue Tracker*, además de describir la implementación realizada en *Trello*. La sección 4.3 aborda el modelo del *CdC*, el cual permite ordenar la lógica y asociar los cambios a elementos específicos, ligado a la implementación de *IT* de la iteración previa. La sección 4.4 trata sobre modelar el flujo de cambio, de tal manera de poder tener visibilidad de las distintas etapas e integrar las herramientas desarrolladas hasta el momento. Finalmente, el ciclo 4.5 busca orientar las mejoras implementadas al proceso de cambio hacia la creación de contenido para realizar sesiones a los clientes de la empresa.

4.1. Habilitando la caracterización del cambio

4.1.1. Situación actual

Al inicio de este ciclo, la situación actual del proceso de cambio corresponde a aquella presentada en la figura 3.5, presente en la página 36. En ella, destaca un objeto implícito asociado a su respectiva alerta: el cambio, ligado a la pregunta “¿Cómo gestionamos el cambio?”

Se recuerda que un objeto implícito es aquel objeto que está parcial o nulamente definido, en otras palabras, no existe suficiente información sobre él. Dada la indefinición de los objetos implícitos, el objetivo de este ciclo es ahondar más sobre la naturaleza del cambio.

Para ello, se decide ser capaz de diferenciar entre distintas sesiones de talleres. Esto permite identificar cuales son las diferencias entre dos sesiones y por ende caracterizar el cambio. En otras palabras se atacan los criterios de aceptación siguientes: F-7 y F-8.

La implementación actual corresponde a una diferenciación por ubicación: un módulo corresponde a una carpeta, esta carpeta posee los siguientes elementos:

- Carpeta *old*: contiene todos los soportes de las sesiones previas.
- Carpeta *recursos*: contiene todos los recursos utilizados dentro de los soportes del módulo, generalmente se trata de medios, figuras o diagramas reutilizables y/o modificables.
- Carpeta *render*: contiene los soportes en formato listo para presentar, es decir *.pdf* y *.pptx*.
- Última instancia de los soportes del módulo, es decir:
 - Cuadernillo en formato *.docx*.
 - Presentación en formato *Google Slides*.

No obstante, este orden solo permite diferenciar entre las sesiones previas, y la instancia actual de los soportes, lo que no entrega la granularidad buscada para poder diferenciar entre dos instancias cualquiera.

4.1.2. Objetivo del ciclo

Diseñar un incremento de solución que permita diferenciar entre todas las instancias de los soportes de los módulos.

4.1.3. Alternativas de solución

Para poder diferenciar entre dos objetos similares, existen dos opciones básicas: que cada objeto posea una característica visible para ser diferenciado del otro, o que exista un tercer elemento que posea información sobre las características de cada objeto y permita distinguirlos.

Por ley del mínimo esfuerzo, se tomará la primera alternativa, y se escogen como características visibles, aquellas de más fácil acceso por el usuario.

4.1.4. Diseño de la solución

Organización del trabajo de implementación

Es importante notar que el trabajo de implementación y diseño realizado para poder versionar las distintas instancias de los módulos fue hecho en colaboración con Franco Cruces, quién es parte del equipo de desarrollo de la empresa. La división del trabajo se realizó en base a pequeñas tareas asociadas a funcionalidades específicas, repartidas de manera equitativa entre ambos desarrolladores, en función de la disponibilidad de tiempo y afinidad con las tareas realizadas. En varias oportunidades se realizó programación de a pares, la cual consiste

en programar en pareja, con un solo computador, lo que permite paralelizar el trabajo, además de revisar el código a medida que se escribe. El trabajo de implementación fue realizado en un 60 % por Franco, y el 40 % restante por el autor de este documento.

Trabajo realizado

Una manera sencilla para poder diferenciar entre dos instancias, es asociar algún tipo de identificador único a cada una de ellas, este representa la versión del módulo.

Por otro lado, se busca definir las características visibles de más fácil acceso por el usuario. Dado que los archivos se encuentran en un directorio de *Drive*, lo primero que se ve es el nombre de cada archivo, y luego de seleccionarlo, se muestra una vista previa de la primera página o slide, para un cuadernillo o presentación respectivamente. Es por ello que sería interesante utilizar dichos espacios para insertar el identificador único.

A modo de resumen, se quiere insertar un identificador único en el nombre y en el interior de cada archivo, características visibles de los archivos.

4.1.5. Implementación

En cuanto al identificador, se decide utilizar la fecha de la última publicación del contenido en formato *AAAAMMDD*, dónde las letras “A” representan los años, las “M” los meses y las “D” los días. Este formato permite ordenar de manera cronológica los archivos a partir de un orden lexicográfico. Además se agrega al inicio una “v” de versión, para finalmente separar el nombre del módulo con el símbolo “~”. El resultado es el siguiente:

$$vAAAAMMDD \sim \text{nombre módulo}$$

A modo de ejemplo un módulo de nombre *Optimizar Flujos de Valor* se transforma en *v20190301 ~ Optimizar Flujos de Valor*. De esta manera es posible diferenciar entre distintas versiones de un módulo, gracias a la fecha.

Si bien se posee un identificador que denota las distintas versiones, es necesario saber dónde ubicarlo de tal manera de ser visible. Se consideran características visibles y de fácil acceso, el nombre de los archivos, particularmente al inicio, de tal manera de lograr el orden cronológico deseado. Además, se agrega el identificador a la esquina inferior izquierda de la primera diapositiva de la presentación, y al pie de página del cuadernillo, en la figura 4.1 es posible ver un ejemplo.

4.1.6. Validación

La manera de validar si la implementación es válida es preguntando directamente al usuario. A partir de la retroalimentación de éste, es posible modificar el diseño y/o implementación

Visibilizar
para entender entre todos

LeanSight

v.20190104

PDF	v.20190304~	/visibilizar para Entender entre Todos - ...	yo
PDF	v.20190304~	/visibilizar para entender entre todos.pdf	yo
P	v.20190304~	/visibilizar para entender entre todos.pptx	yo

Visibilizar para entender entre todos

LeanSight

Visibilizar para entender entre todos.

Objetivo a lograr
Identificar la necesidad de visibilizar colectivamente nuestros pensamientos y quehaceres para lograr entender entre todos.

PENSAMIENTO VISUAL

Para lograr el entendimiento de todos no solo basta con enunciar ideas, conceptos, etc. Este proceso debe ser apoyado a través de la construcción colaborativas de imágenes.

v.20190304

v.20190304

1

Figura 4.1: Ejemplo de implementación de versiones del módulo *v20190304 ~ Visibilizar para entender entre todos*, en la esquina superior izquierda, la primera diapositiva de la presentación, en la esquina inferior izquierda, la última versión publicada, a la derecha, la primera hoja del cuadernillo; encerrado en rojo la ubicación de las versiones.

de la solución, enfocado en generar mayor valor al usuario/cliente. En este caso, distinguir de manera clara diferentes instancias de módulos.

Retroalimentación de los usuarios

La retroalimentación de los usuarios se da en forma de observaciones o preguntas, a partir de las cuales es posible afinar el diseño y/o implementación de la solución. Las observaciones se listan a continuación:

1. ¿Qué ocurre si un usuario publica un mismo módulo más de una vez durante un mismo día? Dado el formato *AAAAMMDD*, éstas serían indistinguibles, por lo que no se cumple con el objetivo del ciclo, es decir distinguir entre versiones de módulos.
2. A raíz de lo anterior, se decide implementar un formato que incluya una mayor granularidad de tiempo, el cual permita discernir entre toda instancia. El lenguaje utilizado dentro de la implementación es JavaScript, éste posee un grado de precisión de milisegundos [26], ¿es necesaria tanta precisión?
3. Por otro lado, el símbolo “~” es el símbolo utilizado al inicio de archivos temporales de documentos de MS Office, documentos que están presentes dentro de las carpetas de los módulos. Es por ello que su uso podría inducir a error, llegando incluso a borrar el archivo versionado, pensando que es un archivo temporal. ¿Es necesario dicho símbolo? ¿Qué cosa podría reemplazarlo?

Modificaciones realizadas a partir de la retroalimentación

En el apartado anterior, se plantean varias preguntas, que permiten mejorar el diseño e implementación de la solución. Las modificaciones se plantearán en el mismo orden lógico presentado, dónde cada punto posee un estado inicial y un nuevo estado. El nuevo estado corresponde al resultado de haber analizado la pregunta y llegado a una implementación que permita solucionarla. Los resultados se muestran en la tabla 4.1.

Observación	Estado inicial	Nuevo estado
1	vAAAAMMDD ~ NM	vAAAAMMDD. THHMiMiSSCMs ~ NM
2	v AAAAMMDD .THHMiMiSSCMs ~ NM	vAAMMDD.THMiMi ~ NM
3	vAAMMDD. THHMiMi ~ NM	vAAMMDD.HHMiMi NM

Tabla 4.1: Tabla comparativa entre los estados de la solución en función de las distintas observaciones. Las letras “NM” corresponden al “Nombre del Módulo”, mientras que las letras “A” corresponden a año, “M” a mes, “D” a día, “.T” es un separador, “H” es hora, “Mi” minuto, “S” segundo, “C” centésima de segundo y “Ms” a milésima de segundo. En rojo, los caracteres que se quitaron, en azul, aquellos que se agregaron.

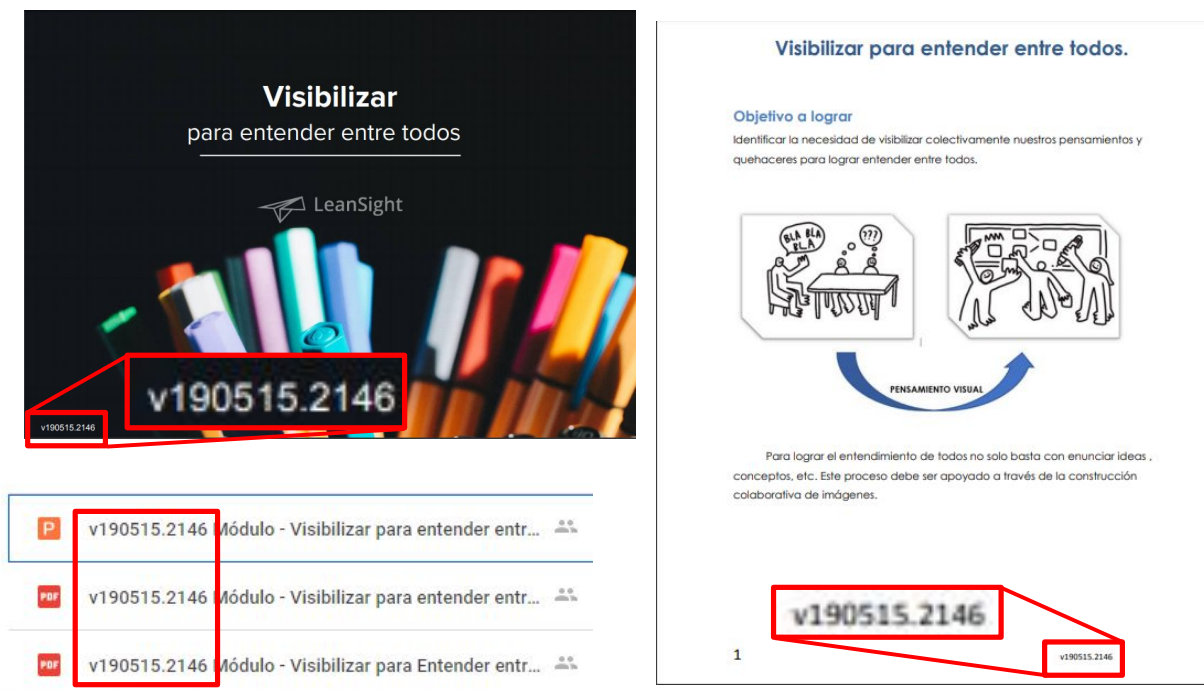


Figura 4.2: Ejemplo de implementación de versiones del módulo *v190515.2146 Visibilizar para entender entre todos*, en la esquina superior izquierda, la primera diapositiva de la presentación, en la esquina inferior izquierda, la última versión publicada, a la derecha, la primera hoja del cuadernillo; encerrado en rojo la ubicación de las versiones.

En la figura 4.2 se encuentra un ejemplo de la versión final de la implementación de versiones.

4.1.7. Aprendizajes

A partir de la implementación de versiones visibles para los usuarios, entendidos como colaboradores, es posible diferenciar entre las distintas instancias de publicaciones de módulos, o sesiones. Esto habilita poder comparar entre las distintas versiones, en particular poder caracterizar los cambios realizados entre dos versiones contiguas.

4.2. Caracterizando el cambio y su flujo

4.2.1. Situación actual

Ahora que está habilitada la caracterización del cambio a través de la implementación de versiones, es necesario identificar la información pertinente para poder realizar dicha caracterización, la cual permite luego compartir y entender de qué se trata un cambio. Además,

es necesario identificar el flujo por el que pasa el cambio. En otras palabras se busca atacar los criterios de aceptación siguientes: F-2, DE-5, DE-6, DE-7, DE-8, DE-9, DE-10, DE-11, DE-12 y DE-13.

La implementación actual es la siguiente: dada una observación realizada por un colaborador, este realiza los cambios que le parecen pertinentes bajo su criterio, y los pone a prueba, sin necesariamente consultar o avisar a los otros diseñadores educacionales.

En otras palabras, dicho modo de operar se hace de forma individual, sin transmitir información al resto del equipo.

4.2.2. Objetivo del ciclo

Este ciclo tiene por objetivo identificar y luego definir las características necesarias para poder transmitir a todos los diseñadores educacionales de qué se trata un cambio o modificación, entendido como una observación de un posible problema.

Además se busca definir el flujo que a partir de un problema bien identificado logra obtener una modificación que da solución a dicho problema.

4.2.3. Alternativas de solución

En lo que respecta a la caracterización y posterior transmisión de un problema observado, existe una solución o arquitectura conocida en el mundo del desarrollo de software: los *Issue Tracking Systems* o *Issue Tracker* (desde ahora en adelante *IT*). Se decide investigar su funcionamiento para ver cómo adaptarlo a la situación presente, es decir los cambios del contenido del *CdC* de la empresa.

Una segunda alternativa, es identificar y listar todos los tipos de cambio que es posible realizar dentro del *CdC*. Si bien esta opción puede ser ineficiente a la hora de entregar información sobre qué características definen el cambio, podría ayudar a entender el funcionamiento del sistema actual. A partir de dicho entendimiento, será posible adaptar el funcionamiento de un *IT* al problema abordado.

En otras palabras, se decide investigar sobre el funcionamiento de un *IT*, para así identificar de qué manera se caracteriza y comparte la información pertinente a un problema, para luego adaptarla en función de la información recopilada al listar todos los posibles cambios del *CdC*.

4.2.4. Diseño de la solución

En este caso, los incidentes son las observaciones realizadas por los diseñadores educacionales o los facilitadores con respecto a las sesiones realizadas o módulos diseñados.

Basándose en el funcionamiento del *IT*, es necesario denotar el contexto de una observación, la cual permite sustentar el cambio a realizar. En otras palabras, se requiere:

- La instancia o versión en que ocurrió el problema
- Una breve descripción de lo ocurrido.
- La severidad de lo ocurrido, de tal manera de poder priorizar luego. A modo de ejemplo, si hay un error a nivel estético, no es necesario arreglarlo de manera urgente, no así el caso en que el usuario no sea capaz de ingresar a su cuenta.
- Datos suficientes para poder reproducir el error, con tal de poder verificar si este persiste o no.
- En qué lugar preciso ocurrió el error.

Además dicha información debe poder ser accesible por todo el equipo, generando así entendimiento común.

Con el fin de identificar la naturaleza de un cambio o modificación, se parte de casos concretos. Para ello, se identifican todos los posible tipos de cambio, es decir toda modificación de cualquiera de los soportes, y se analiza que tipo de propagación genera. En otras palabras, se busca identificar el origen de los cambios y de qué manera repercuten en el resto de los recursos.

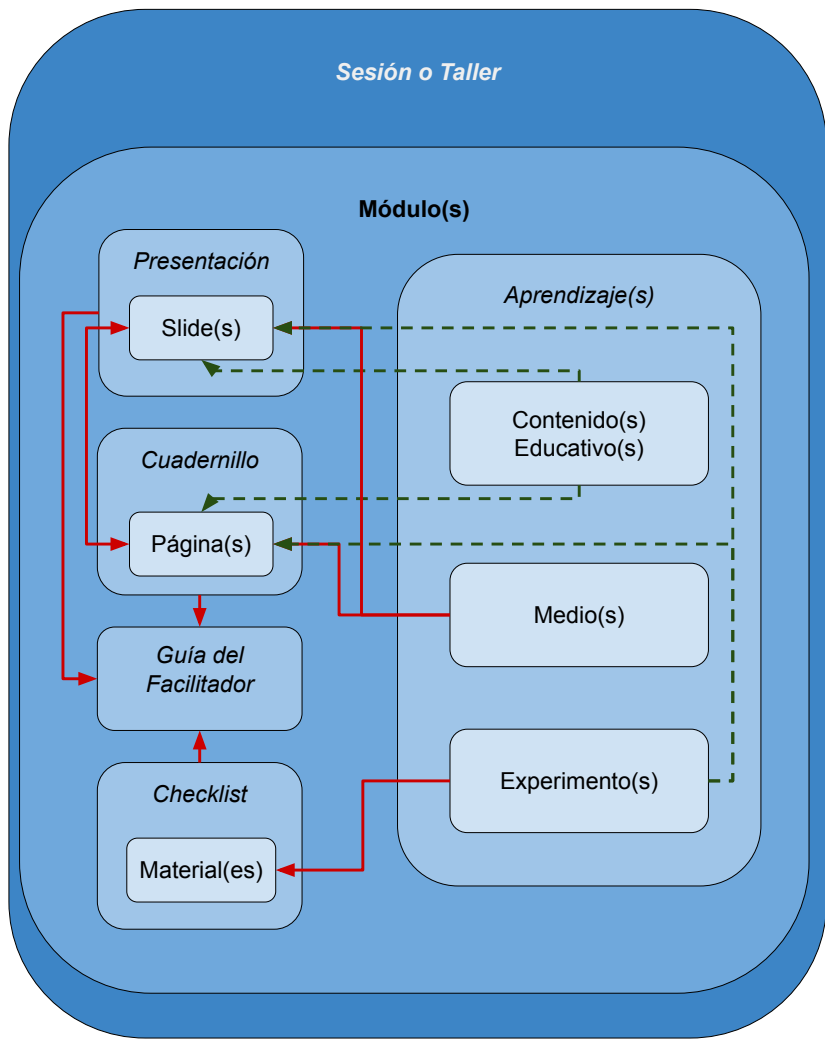
Antes de continuar con los resultados, es necesario hacer un recordatorio de ciertos elementos del modelo de la situación actual:

- Una sesión o taller está compuesto de uno o más módulos.
- Un módulo está compuesto de aprendizajes y soportes.
- Los aprendizajes consisten en contenido educativo, medios y experimentos.
- Los soportes son la presentación, el cuadernillo, el checklist y la guía del facilitador.
- Los soportes representan la manera concreta de generar entendimiento sobre los aprendizajes a los aprendices. En otras palabras, los aprendizajes se traducen en los distintos soportes de un módulo.

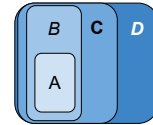
A partir de lo anterior, y del desglose de los distintos tipos de cambio y su propagación, se llega a los siguientes resultados:

- Un cambio en el contenido educativo o en un medio podría generar cambios en los slides de una presentación y/o las páginas de un cuadernillo.
- Un cambio en un experimento podría generar cambios en las slides de una presentación, las páginas de un cuadernillo y/o el checklist del módulo.
- Un cambio en las slides de una presentación podría generar un cambio en las páginas del cuadernillo.
- Los cambios ocurridos en un elemento **A** contenido por otro elemento **B**, implican que el elemento **B** también cambia. A modo de ejemplo, si cambió una slide, entonces la presentación que la contiene cambia también, lo mismo ocurre en el caso de modificar un módulo, la sesión que la contiene también se ve afectada por el cambio.

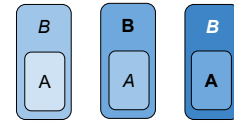
A modo de resumen de los resultados, se presenta la figura 4.3.



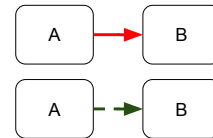
Simbología



A compone a B, B compone a C, C compone a D.



A compone a B, por lo que si se modifica A, entonces B también es modificado.



Si se modifica A, entonces B también podría ser modificado (propagación del cambio).

Nota: no existe diferencia entre la flecha continua roja y la línea segmentada verde, están ahí para evitar confusión entre el origen y destino al cruzarse las flechas.

Figura 4.3: Diagrama resumen de los tipos de cambio y su propagación entre los distintos elementos del *CdC*.

De lo anterior es posible concluir que los cambios surgen desde un cambio en un aprendizaje, el cual se propaga a través de los distintos soportes para mantener la consistencia. En otras palabras, la mejora de los contenidos pedagógicos se ve reflejada en cambios a los elementos y estos se propagan por consistencia de los datos. Concretamente, es necesario tomar en cuenta tanto la razón del cambio, cómo se traduce a los soportes y de qué manera se propaga.

4.2.5. Alternativas de implementación

Para lograr caracterizar el cambio, vimos en la sección anterior que se necesita una razón que lo sustente, además de indicar de qué manera se traduce dentro de los soportes, sin olvidar cómo este se propaga. A modo de implementación, se propone generar una bitácora de cambios visible y accesible por todo el equipo donde poder denotar la razón, soporte y propagación del cambio.

En lo que respecta visibilizar la información, se plantean dos alternativas. La primera contempla una planilla de cálculos de *Drive*, accesible por todo el equipo. La segunda utiliza una herramienta de gestión del trabajo en equipo llamado *Trello* [22]. Esta consiste en una serie de estados que denotan un flujo, y una serie de tareas que pasan de un estado a otro. Esto es representado por distintas listas en forma de columnas (flujo), y tarjetas que pueden moverse de una columna a otra (tareas). Esto se explicará más en detalle en la sección 4.2.6, en particular en la figura 4.7.

En lo que respecta la información necesaria, es la misma para ambas alternativas, pero la forma de representarla sería distinta. A continuación se muestran ambas implementaciones.

4.2.6. Implementación

Planilla de Cálculos - *Google Spreadsheet*

A modo caracterización del cambio, se diseñaron plantillas o templates con toda la información pertinente para transmitir el problema a otro diseñador educacional. La idea es rellenar manualmente el formulario en función del cambio realizado. A continuación se presentan capturas de pantalla de templates de bitácoras de cambio asociados a los distintos cambios posibles. Para cada plantilla, se destacan los recuadros “Por qué cambió?”, además del tipo de cambio y su propagación, es decir los elementos necesarios para caracterizar el cambio.

En la figura 4.4 es posible ver un ejemplo de las planillas asociadas a los cambios realizados a nivel de taller o sesión y módulo.

Taller		¿Qué módulo?		¿En qué taller se hallaba el módulo anteriormente?					
Quién	Cuándo	Nombre Taller	Dónde Versión	Link al taller	Tipo de Cambio	Nombre	Versión	Nombre Taller	Versión
	¿Por qué cambio?	Presentación	Implicancias de cambio en Cuadernillo	Si	Checklist	Si	Si	¿De qué manera? Cuadernillo	Checklist
		No	Guía Facilitador	No	Checklist	Si	Si	¿De qué manera? Cuadernillo	Checklist

Módulo		¿Qué Aprendizaje?		¿En qué módulo se hallaba el aprendizaje anteriormente?					
Quién	Cuándo	Nombre Módulo	Dónde Versión	Link al módulo	Tipo de Cambio	Nombre	Versión	Nombre	Versión
	¿Por qué cambio?	Presentación	Implicancias de cambio en Cuadernillo	Si	Checklist	Si	Si	¿De qué manera? Cuadernillo	Checklist
		No	Guía Facilitador	No	Checklist	Si	Si	¿De qué manera? Cuadernillo	Checklist
					Quitar Aprendizaje				

Figura 4.4: Ejemplo de las plantillas para taller/sesión y módulo por utilizar en la bitácora de cambios realizada en *Google SpreadSheet*. Encerrado en verde está el elemento modificado, en magenta la razón del cambio, además los colores rojo y azul representan la implicancia de la propagación del cambio, en rojo en caso de propagación, azul cuando no la hay.

En las figuras 4.5 y 4.6 es posible ver un ejemplo de los templates de la bitácora de cambios de un medio/contenido o actividad respectivamente.

Medio						
Quién	Cuándo	Dónde			Slide(s)	Tipo de Cambio al Medio
		Presentación	Versión	Link a la presentación		Agrega
	¿Por qué cambió?	Implicancias de cambio en		¿De qué manera?		
		Presentación	Guía del Facilitador	Presentación	Guía del Facilitador	
		Si	Si			

Información						
Quién	Cuándo	Dónde			Slide(s)	Tipo de Cambio a la Información
		Presentación	Versión	Link a la presentación		Reformulación
	¿Por qué cambió?	Implicancias de cambio en		¿De qué manera?		
		Presentación	Guía del Facilitador			
		No	No			

Figura 4.5: Ejemplo de las plantillas para medios e información por utilizar en la bitácora de cambios realizada en *Google SpreadSheet*. Encerrado en verde está el elemento modificado, en magenta la razón del cambio, además los colores rojo y azul representan la implicancia de la propagación del cambio, en rojo en caso de propagación, azul cuando no la hay.

Actividad						
Quién	Cuándo	Dónde			Página(s)	Tipo de Cambio a la Actividad
		Cuadernillo	Versión	Link al cuadernillo		Reformulación
	¿Por qué cambió?	Implicancias de cambio en			¿De qué manera?	
		Presentación	Guía del Facilitador	Checklist	Presentación	Checklist
		Si	No	Si		

Figura 4.6: Ejemplo de las plantillas para actividades por utilizar en la bitácora de cambios realizada en *Google SpreadSheet*. Encerrado en verde está el elemento modificado, en magenta la razón del cambio, además los colores rojo y azul representan la implicancia de la propagación del cambio, en rojo en caso de propagación, azul cuando no la hay.

Trello

La herramienta *Trello* permite hacer uso de la metodología de Tablero Kanban [10]. Como se dijo anteriormente, esta consiste en una serie de estados que denotan un flujo, y una serie de tareas que pasan de un estado a otro. Esto es representado por distintas listas en forma

de columnas (flujo), y tarjetas que pueden moverse de una columna a otra (tareas).

Cabe resaltar que las columnas o estados posibles poseen un límite sobre la cantidad de tarjetas que pueden poseer, también denominado “Trabajo a medio hacer” o *Work in Progress*, a partir de ahora *WiP*.

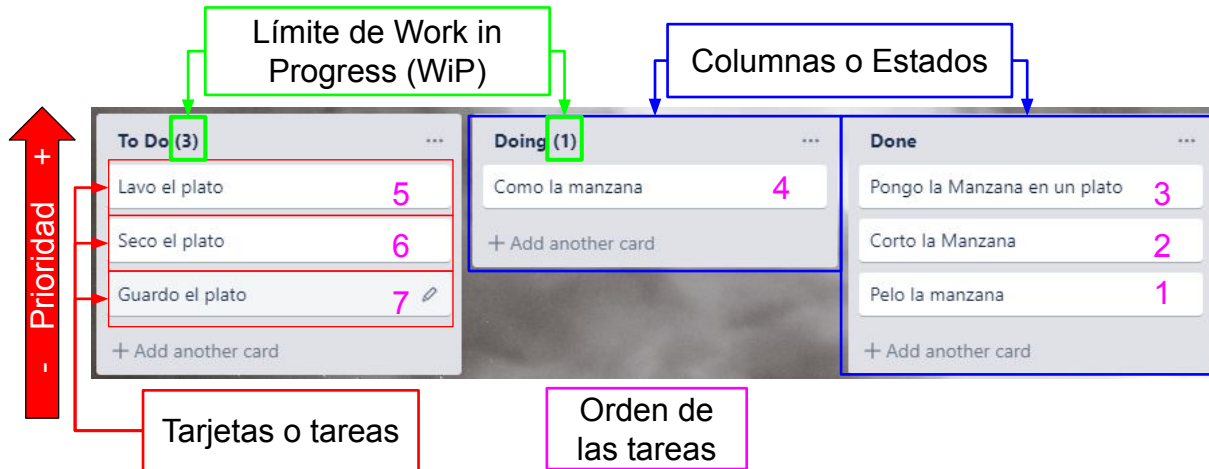


Figura 4.7: Ejemplo de los pasos para comer una manzana reflejados en un tablero Kanban con todas sus componentes: encerradas en rojo las tarjetas o tareas; en azul, las columnas o estados; en verde, el límite de *WiP*; en magenta, el orden de las tareas. La prioridad de las tareas por hacer está denotada por el orden de estas, más arriba es más prioritario.

Para este caso se definen las siguientes tipos de incidentes:

- Issue: Se refiere a todo tipo de problema que no está del todo determinado, es decir necesita de experimentos para determinar si es posible técnicamente, o de conversar para llegar a acuerdo con el resto de los diseñadores.
- Cambio: se refiere a un issue que está suficientemente delimitado para llegar a ser concreto. En regla general, un issue se transforme en un cambio cuando existe consenso entre todos los diseñadores educacionales sobre lo que ha de ser realizado.

En el caso de esta implementación, se hace frente a incidentes o *issues*, los cuales requieren de experimentación, para luego transformarse en cambios concretos. Dada la dependencia entre issues y cambios, se llega al siguiente flujo [29]:

1. Issue - Backlog: lugar inicial de toda tarjeta correspondiente a un issue.
2. Issues - Listo para experimentar: una vez que las tarjetas pasan a este estado, el trabajo que representan ha sido comprometido, es decir se debe realizar.
3. Issues - Experimentando: corresponde al trabajo que se está realizando en ese momento.
4. Issues - Diseño de cambios asociados: Aquí están las tarjetas que ya presentan resultados de experimento, por lo que debiesen diseñarse cambios asociados.
5. Issues - Listo para realizar cambios: Aquí se encuentran las tarjetas de issue que poseen cambios definidos, por lo que están a la espera de que todos los cambios asociados

terminen. Además, tienen asociadas las tarjetas de cambio que deben ser realizadas para considerarse como hechas.

6. Issues - Done: corresponde a las tarjetas de issue que ya vieron todos sus cambios asociados realizados.
7. Cambios - Backlog: lugar inicial de toda tarjeta correspondiente a un cambio. Ojo que los cambios están asociados a issues, en otras palabras los cambios son las acciones a realizar para poder resolver un issue.
8. Cambios - To Do: Toda tarjeta de cambio que se encuentra aquí está comprometida.
9. Cambiando: cambios siendo realizados en dicho momento.
10. Check: los cambios realizados deben ser verificados por otro diseñador educacional, de tal manera de corroborar que las modificaciones sean correctas.
11. Cambiado (Done): aquellos cambios que ya fueron realizados y verificados.

En la figura 4.8 es posible ver las etapas que sigue una tarjeta de issue dentro del *IT* implementado en Trello.

4.2.7. Validación

A continuación se presentan las distintas observaciones y/o retroalimentaciones de los usuarios, las cuales pueden estar expresadas en forma de preguntas. Luego, se discuten las observaciones para finalmente presentar las modificaciones por realizar.

Retroalimentación de los usuarios

1. La bitácora de cambios implementada en *Google Spreadsheet* no está adaptada a los procesos de la empresa, por lo que no cumple con la característica de calidad de adaptación al proceso actual. ¿Qué aprendizaje es posible sacar de tal situación? A diferencia de la implementación en *Google Spreadsheet*, Trello es una herramienta utilizada por el equipo, es por ello que no hay curva de aprendizaje para aprender a usarla, lo que significa que está más adaptada a los procesos de la empresa.
2. Dada la caracterización de los cambios en el *IT* implementado en Trello, ¿de qué manera es posible llegar al lugar específico donde se encuentra el cambio/issue caracterizado?
3. Si bien la solución está implementada, los usuarios son reticentes a usarla. ¿Qué característica posee la solución que la hace poco utilizable?
4. ¿Qué falta para poder utilizarla?

Discusión a partir de la retroalimentación

A continuación se presentan respuestas

1. La solución implementada en *Google Spreadsheet* finalmente no fue utilizada, pero permitió tener un primer acercamiento a la manera en que se podría caracterizar el cambio

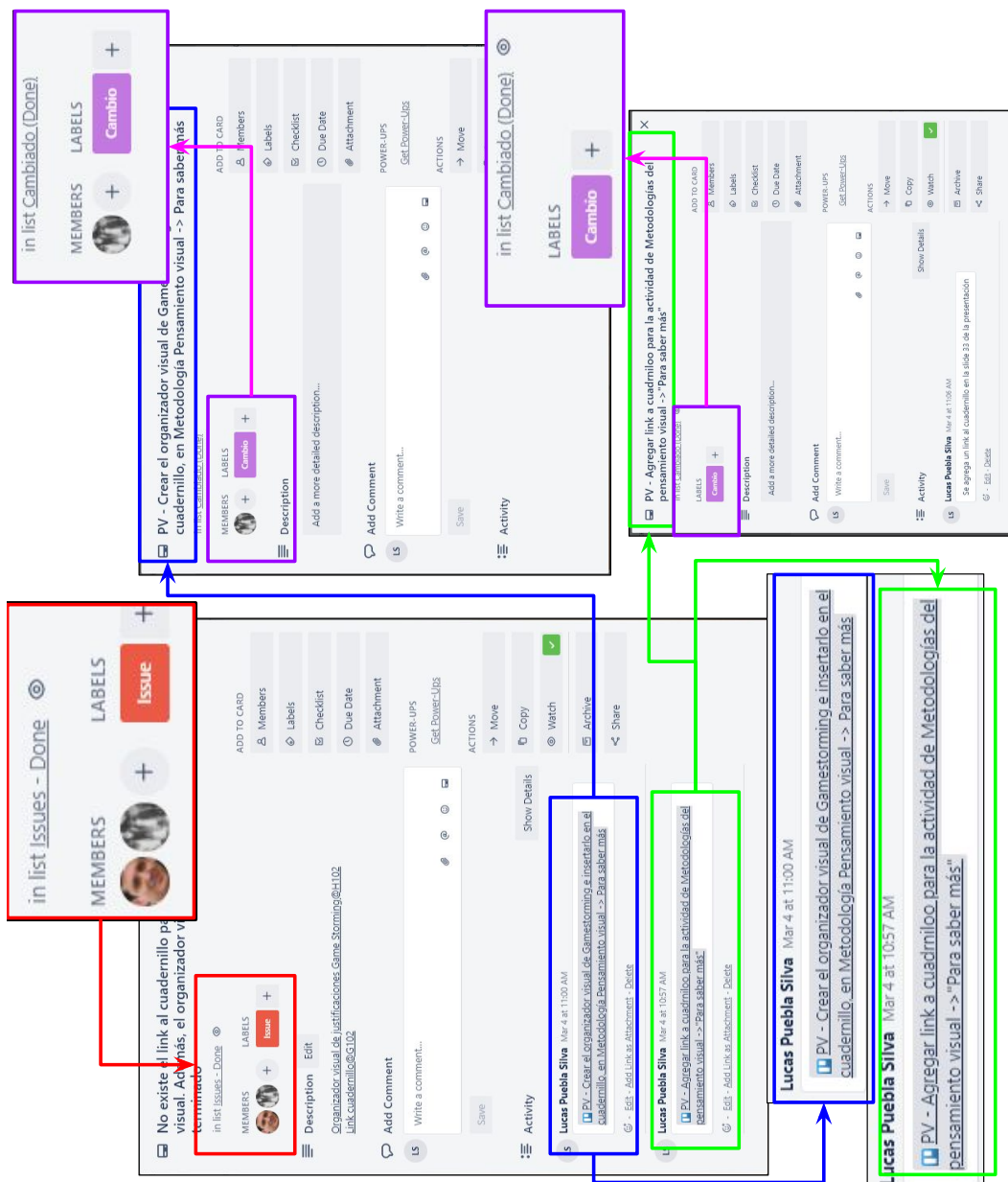


Figura 4.8: Ejemplo del flujo seguido desde un Issue a un Cambio dentro de la implementación del IT en Trello, los recuadros del mismo color corresponden a ampliaciones de la captura de imagen, los colores verde y azul denota un cambio asociado distinto. Los colores rojo y morado, corresponde a el tipo incidente, en rojo issues y en morado cambios. Nótese que al haberse realizado ambos cambios, dado que se encuentran en el estado Done (recuadro morado), el issue también está en Issues - Done (recuadro rojo).

adaptado al contexto de la empresa (ver sección 3.4.3). Es decir se logró identificar la información necesaria para caracterizar el cambio.

2. Actualmente no existe un identificador único para los distintos elementos o componentes de los recursos. La falta de tal identificador dificulta poder acceder a los componentes de manera directa. Esto implica que es costoso en tiempo para los diseñadores educacionales el realizar una modificación a partir del *IT*.
3. La solución no se utiliza dado que no calza con la implementación actual, por lo que demanda trabajo extra a los colaboradores. Rompe con la característica de calidad de adaptación al proceso actual (ver sección 3.4.3).

Siguientes modificaciones a partir de la retroalimentación

Para poder usar el *IT*, es necesario simplificar su uso para los usuarios. En otras palabras, es necesario reducir la complejidad de encontrar la ubicación exacta de los cambios o issues por realizar. Además, sería posible integrar su uso dentro de otra herramienta más adaptada a los procesos de la empresa. Otra característica importante es hacer viable el uso de la herramienta enfocándose en la experiencia usuario.

4.2.8. Aprendizajes

Luego de realizado este incremento de solución, es posible caracterizar los cambios, además de poseer un soporte accesible por todo el equipo que modela el flujo de los incidentes. No obstante, la solución implementada es costosa en tiempo dada la falta de un acceso directo a los cambios o issues que requieran de modificaciones. Para poder facilitar esto, el siguiente paso está en modelar de manera concreta los módulos, de tal manera de poder ubicar en qué lugar se ha de realizar el cambio.

4.3. Modelando los módulos

4.3.1. Situación actual

A partir de los incrementos de solución previamente descritos, es posible:

- Caracterizar los incidentes, correspondiente a issues y cambios asociados.
- Conocer el ciclo de vida, o flujo, por el que pasan los cambios e issues.
- Visibilizar y dar acceso a todo el equipo a la información pertinente correspondiente a los cambios e issues a través del *Issue Tracker*.

No obstante, es necesario poder entender la lógica detrás de cada módulo. Esto permite entender el propósito detrás de cada cambio o issue. Para ello, se propone modelar los mó-

dulos, para así visibilizar la progresión lógica, o relato, y habilitar el entender y gestionar sus contenidos.

En otras palabras se quiere atacar los siguientes criterios de aceptación: F-1, DE-1, DE-2, DE-3 y DE-4.

Actualmente, el relato de los módulos se compone de una estructura donde la mayoría de los aprendizajes posee una presentación del problema, seguida de un experimento que permite evidenciarlo, luego de lo cual se presenta un principio de solución que viene a solucionar el problema evidenciado en el experimento. Si bien esta estructura lógica se repite, existen contenidos que no la siguen, lo que dificulta en gran medida entender el relato, es decir, la progresión lógica de un módulo. Además es necesario considerar que se utilizan tanto la presentación como el cuadernillo, soportes relacionados entre sí, a modo de apoyo para transmitir los aprendizajes.

4.3.2. Objetivo del ciclo

Modelar los módulos de tal manera de visibilizarlos a todo el equipo y poder denotar y entender la lógica de cada uno, en particular las relaciones entre soportes. Además se requiere poder asociar los incidentes (issues y cambios) con elementos específicos de los módulos, de tal manera de tener un centro de control desde el cual iniciar y terminar las modificaciones del contenido del *CdC*.

4.3.3. Alternativas de solución

A modo de solución, se busca modelar un módulo a partir del modelo detallado de la situación actual presente en el anexo B. Para mantener un índice de calidad, se priorizan soluciones que se adapten a los procesos existentes o herramientas disponibles (ver sección 3.4.3), además de tener en cuenta la sencillez y usabilidad de la solución (ver sección 3.4.2).

Más que alternativas de solución, este ciclo trata de ajustar el modelo detallado de la situación actual presente en el anexo B para poder modelar un módulo. Esto fue realizado con el resto del equipo presente de tal manera de llegar a un modelo que funcionase y que permitiese denotar las relaciones entre los distintos elementos.

4.3.4. Diseño de la solución

El diseño de la solución consiste en comprobar si el modelo detallado de la situación actual, específicamente los elementos que conforman un módulo, permiten cumplir con el objetivo del ciclo, es decir visibilizar la lógica o relato de un módulo, asociar los distintos soportes, y permitir asociar incidentes a elementos específicos del módulo. Para ello se toma el modelo, y se intenta modelar el módulo “Visibilizar para entender entre todos” de la manera más

rápida posible, con el apoyo de parte del equipo, con tal de validar el funcionamiento de dicho modelo.

4.3.5. Alternativas de implementación

Para llevar a cabo este ciclo, es posible utilizar una gran variedad de herramientas. Se destacan dos y se listan los pro y los contra de cada una, para poder compararlas.

La primera herramienta es el simple uso de post-its dentro de una pizarra blanca y la ayuda de plumones, a partir de los cuales modelar un módulo. Para ello se requiere simplemente una cantidad suficientes de post-its diferenciables que permitan entender el modelo, en otras palabras colores, orientaciones y/o formas distintas. La ventaja de esta implementación, es que es sumamente barata, fácil de cambiar y permite la interacción de varias personas a la vez. Por otro lado, esta implementación utiliza parte del espacio de las pizarras, el cual es limitado y es muy utilizado, por lo cual la persistencia de la implementación se ve afectada. A modo de contra-medida, es posible tomar una foto del estado final, con lo que se perdería gran parte de los pro, dado que una foto es estática.

La segunda herramienta corresponde a una planilla de cálculo, lo cual necesita de un computador con acceso a internet para usar *Google Spreadsheet* o eventualmente acceso a *Microsoft Excel*. La ventaja de esta implementación es que ocupa un espacio ínfimo ⁹, es editable en todo momento, incluso por varios usuarios a la vez en el caso de *Spredsheet*. No obstante, la herramienta requiere un poco más de tiempo para ser implementada, necesita de mayor coordinación en el caso de que varios usuarios la editen al mismo tiempo, y en caso de tener demasiada información, se vuelve difícil ver la totalidad del contenido a menos de tener un soporte adecuado (como una pantalla con alta resolución y lo suficientemente grande para poder leer todos los elementos).

A partir del análisis de ambas implementaciones posibles, se decide probar el modelo utilizando post-its, una pizarra, sharpies y plumones, para luego tomar una foto y luego digitalizar el modelo utilizando un *Google Spreadsheet*. Esto conserva tanto la flexibilidad de la primera solución como la persistencia y uso eficiente del espacio de la segunda solución.

Es posible notar que han existido propuestas que resuelven este problema [6], pese a que lamentablemente a la fecha no existe una implementación de ello.

4.3.6. Implementación

La implementación consiste en la actividad siguiente: se reúnen 3 diseñadores educacionales con el autor de este documento y se les pide adaptar el modelo detallado de la situación actual presente en el anexo B a uno que logre modelar correctamente un módulo. Para ello, se toma como ejemplo el módulo “Visibilizar para entender entre todos”, dado que es uno de

⁹Una planilla *Excel* pesa del orden de 1MB, mientras que cualquier producto de Google, tal como *Docs*, *Spreadsheet* o *Slides* no ocupa espacio en *Drive*.

los módulos más maduros, es decir, al cual se realizan una menor cantidad de cambios en comparación a otros módulos. Los pasos a seguir son los siguientes:

1. Se proyectan tanto la presentación como el cuadernillo en una pantalla visible a todo el equipo.
2. Se pega un post-it con el nombre de cada uno de los elementos que componen un módulo hasta el momento. Inicialmente, se trata de aprendizajes, presentación, cuadernillo, experimentos y medios.
3. Para el elemento siguiente dentro del relato o lógica del módulo, partiendo por la primera diapositiva y yendo tan lejos como sea necesario para validar el modelo, se realiza lo siguiente:
 - (a) En un post-it, se nombra el elemento y se pega bajo la columna correspondiente, en una nueva fila. En otras palabras el relato avanza por filas de arriba hacia abajo.
 - (b) Si no existe una columna que pueda describir el elemento, entonces se le pide a los participantes discutir sobre la manera en que se pueden modelar, para agregar una nueva columna en forma de post-it. Con este nuevo post-it, se vuelve al paso 2.
 - (c) Si por el contrario, existe una categoría que define al elemento, y este hace referencia a algún otro, por ejemplo una diapositiva que hace referencia a una página del cuadernillo o a un medio, se pone un post-it con un nombre representativo en la columna correspondiente y en la misma fila que el post-it anterior. En otras palabras, dos post-it en una misma fila denota que están relacionados entre sí.
 - (d) Si no existe un soporte relacionado, se vuelve al paso 3 con el elemento siguiente.

Es posible notar que para llevar a cabo la implementación, sólo fue necesario pasar por 13 elementos del relato, separados en 6 filas. A continuación se describen las categorías agregadas:

1. Evaluaciones: cada aprendizaje debiese ser evaluado de alguna forma para asegurar que el aprendiz adquirió dicho aprendizaje.
2. Estructura: dado que los componentes de un módulo intentan seguir una lógica de vivencia del problema, presentación del principio de solución, y vivencia de la misma situación problemática tomando en cuenta el principio de solución, se requiere identificar la estructura.
3. Nombre: cada componente debiese tener un nombre, de tal manera de poder identificarlo.
4. Materiales: se refiere a todos los materiales necesarios para presentar un módulo. Por regla general, se necesita sharpies y post-its para cada aprendiz, además de parlantes y alguna manera de proyectar la presentación. Existen más materiales, en particular para ciertos experimentos.
5. Actividades: se refiere al conjunto de soportes que permiten entregar contenido a los aprendices, es decir presentación, cuadernillo, experimentos y medios.
6. Problema: representa un problema por solucionar. Este problema se traduce en alguna modificación luego. A modo de ejemplo, se podría decir que el cuadernillo y la presentación no son consistentes, dado una imagen distinta.

7. Solución: representa la manera de solucionar un problema. Siguiendo el mismo ejemplo de arriba, la solución sería cambiar la imagen para que sea consistente.
8. Experimento: se refiere a aquellos problemas que requieren de un experimento para saber cómo ser solucionados. En otras palabras poseen un grado de incertidumbre. A modo de ejemplo, se podría buscar experimentar sobre uso de vocabulario para denotar una idea. Luego de probar distintos términos en sesiones distintas, sería posible ver cuál es la mejor solución.
9. Aprendizajes generales, específicos y detalle: se refiere a distintas clasificaciones de aprendizajes, donde los aprendizajes generales están compuestos por aprendizajes específicos, y a su vez estos están compuestos por aprendizajes detalle.

En la figura 4.9, es posible ver el resultado de la actividad.

Acto seguido, se procede a digitalizar el modelo utilizando *Google Spreadsheet*. a partir de lo cual se obtiene el resultado presentado en la figura 4.10. Nótese que se agrega un identificador para poder referenciar un elemento del relato, además del nombre del módulo modelado.

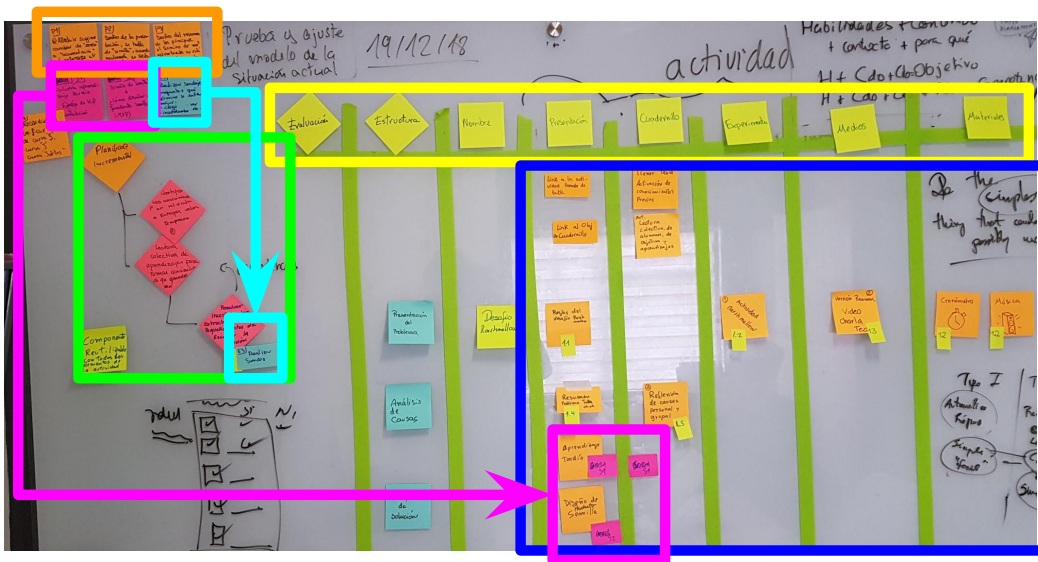
4.3.7. Validación

La validación se realiza a partir del uso de la herramientas por los usuarios, quienes entregan retroalimentación, la cual será planteada a continuación en formato de preguntas, que orientan el desarrollo para obtener una herramienta más usable.

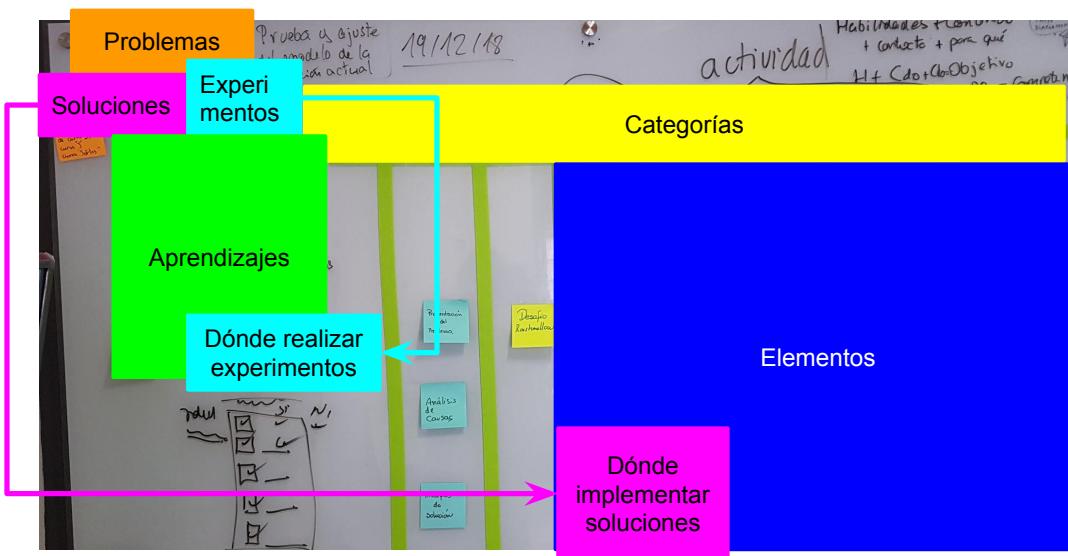
Retroalimentación de los usuarios

A raíz del uso de la herramienta, surgen las siguientes preguntas:

1. A diferencia de la implementación en baja tecnología, en la cual cada forma, color u orientación de los post-its utilizados denotaba los distintos elementos, la implementación digital solo posee columnas y filas. ¿De qué manera sencilla es posible diferenciar los elementos dentro de la implementación digital del *MoME*?
2. Los aprendizajes son un objeto indefinido, del que no se posee demasiada información, por lo que es difícil poder clasificar en cuanto a los tipos de aprendizaje y de qué manera se componen entre ellos. ¿De qué manera es posible evitar tener que clasificar, manteniendo los aprendizajes?
3. Existen elementos dentro del relato que son más importantes que otros, en otras palabras, son los focos de atención. A modo de ejemplo, al realizar una actividad en el cuadernillo, la presentación hace alusión al uso de éste, pero el cuadernillo es el foco en ese momento. ¿De qué manera es posible denotar distintos niveles de importancia entre los soportes?
4. Las palabras “Problema” y “Experimento” son demasiado amplias para denotar las posibles modificaciones, además de que es posible confundir entre un actividad de experimento y un experimento para saber qué modificación realizar. ¿Qué palabras podrían



(a) Foto con recuadros del estado final del Modelo de Módulos Educacional o *MoME*



(b) Representación de los distintos elementos del estado final del Modelo de Módulos Educacional o *MoME*

Figura 4.9: Comparación entre la foto con recuadros y una representación del estado final del Modelo de Módulos Educaionales o *MoME*, luego de realizada la implementación para adaptar el modelo detallado de la situación actual a un módulo real, con recuadros que denotan los distintos elementos. Nótese que la misma imagen fue usada para las figuras 4.9(a) 4.9(b).

[Nombre módulo]						
Aprendizaje (*)						
ID (***)	General	Específico	Detalle	Evaluación	Estructura	Nombre

Actividades (**)					
Presentación	Cuadernillo	Experimento	Medios	Materiales	

Figura 4.10: Implementación en *Google Spreadsheet* del *MoME*, a partir de los datos obtenidos en la actividad con baja tecnología representada en la figura 4.9.

ser utilizadas para mantener el significado, evitar confusiones y ser más preciso con el vocabulario?

5. Los aprendizajes están enfocados en entregar a los aprendices los conocimientos necesarios para que ellos puedan hacer un cambio de paradigma. ¿Cómo denotar el paradigma que sustenta el aprendizaje?
6. Ahora que se tiene un modelo de los módulos educacionales, que era una necesidad para poder visibilizar el relato utilizando el *Issue Tracker* implementado (ver sección 4.2), ¿De qué manera es posible conectar ambas herramientas?

Modificaciones realizadas a partir de la retroalimentación

A partir de las preguntas realizadas en la sección anterior, se listan a continuación las modificaciones realizadas, para finalmente mostrar una captura de pantalla del estado final del *MoME*:

1. Con respecto a 1, para poder diferenciar los distintos elementos, se utilizan los mismos colores que se utilizaron durante la implementación en baja tecnología dentro de la implementación en *Google Spreadsheet*. En otras palabras se aplica un código de colores.
2. Con respecto a 2, la clasificación de los distintos aprendizajes se desecha, por lo que los aprendizajes ya no son clasificados en distintos tipos, sino que sólo existe un tipo de aprendizaje.
3. Con respecto a 3, se utiliza un mismo color con tonalidades variables, desde más claro y menos importante a más oscuro y de mayor importancia o foco.
4. Con respecto a 4, la palabra “Problema” se refiere a una modificación específica, por lo que se reemplaza por la palabra “Cambio”. En cuanto a la palabra “Experimento”, se reemplaza por la palabra “Issue”, que en español significa “Incidente”. De esta manera se utiliza un vocabulario más preciso que evite confusiones.
5. Con respecto a 5, se agrega una columna de “Paradigma” al lado de la columna de “Aprendizajes”, lo cual implica que un paradigma sustenta uno o más aprendizajes.

6. Con respecto a 6, se vinculan ambas herramientas, utilizando hipervínculos que permiten llevar desde el *MoME* al *Issue Tracker* y de vuelta. Además de esta manera es posible denotar un incidente o cambio ligado a un elemento específico del relato. Para ello se implementó un componente o *Add-on* utilizando *Google Apps Scripts*, llamada *Trello For MoME*.

En la figura 4.11 es posible ver una captura de pantalla de la versión actual del *MoME*.

4.3.8. Aprendizajes

En esta sección se logra modelar los módulos educacionales, dando lugar al *MoME*. Esto habilita el visibilizar el relato de los módulos, lo que a su vez habilita el uso del *Issue Tracker*, dado que es posible desglosar los elementos del relato, y por ende precisar qué modificación corresponde a qué elemento.

4.4. Modelando el proceso de cambio

4.4.1. Situación actual

A partir de los resultados de los ciclos anteriores, es posible:

- Caracterizar los incidentes
- Visibilizar el estado de estos dentro del flujo del *Issue Tracker*
- Visibilizar y dar acceso a todo el equipo a la información pertinente correspondiente a los cambios e issues a través de *Issue Tracker*.
- Visibilizar el relato lógico de los módulos a través del Modelo de Módulos Educacionales o *MoME*, el cual está conectado con el *Issue Tracker*, de tal manera de poder asociar incidentes a elementos específicos de un módulo.

La siguiente etapa trata de modelar el proceso de cambio actual para luego definir un proceso de cambio objetivo a alto nivel. En otras palabras se busca atacar los siguientes criterios de aceptación: DE-21, DE-22, DE-23, DE-24, DEF-1, DEF-2 y F-3.

Nótese que si bien los criterios de aceptación citados hacen alusión a algún tipo de soporte, es necesario tener un flujo a partir del cual basarse para poder implementar dicho soporte.

4.4.2. Objetivo del ciclo

Establecer una visión a alto nivel del proceso de cambio, la cual permita integrar las distintas herramientas/implementaciones llevadas a cabo hasta ahora.

Sección	Aprendizaje	Paradigmas previos	Estructura	Nombre
	Aprendizaje general del módulo	Paradigma previo que necesita del aprendizaje general (opcional)		
Nombre Sección				
	Nombre aprendizaje	Paradigma previo que necesita del aprendizaje		
		Issue		
		Cambio		

Actividades Pedagógicas				
Presentación	Cuadernillo	Experimento	Medios (*)	Preguntas de reflexión
Elemento menos importante	Elemento poco importante	Elemento importante	Elemento Foco	¿Pregunta?
Cambio			Ejemplo de Medio	
Issue				

(a) Parte 1 de 3 del *MoME*, las partes van contiguas de izquierda a derecha.

(b) Parte 2 de 3 del *MoME*, las partes van contiguas de izquierda a derecha.

Materiales			Evaluación
Material 1	Material 2	Material 3	Evaluación

Código Issues (**)	Issue
	Cambio

(*) Los medios deben ir linkeados al medio correspondiente, valga la redundancia
 (**) Los Issues y cambios deben tener el link al issue dentro del trello "LS Toolkit - Issue Tracker"
 (***) Las tarjetas deben tener el link a la celda del modelo que están referenciando. Se obtiene con Click Derecho en la celda y "get link to this cell"

(c) Parte 3 de 3 del *MoME*, las partes van contiguas de izquierda a derecha.

(d) Nomenclatura presente en el *MoME*.

Figura 4.11: Captura de pantalla de la versión actual del *MoME*. Nótese que es posible observar todas las modificaciones realizadas a partir de las retroalimentaciones de los usuarios. Nótese que las figuras 4.11(a), 4.11(b) y 4.11(c) corresponden a la misma fila, pero por temas de espacio, se separaron en tres partes. Con respecto a la figura 4.11(d), corresponde a parte de la nomenclatura, además de ciertos detalles de implementación que facilitan la interacción entre los soportes del *MoME* e *Issue Tracker*.

4.4.3. Alternativas de solución

Basándose en el conocimiento de sistemas de gestión de versiones provenientes desde el mundo de las ciencias de la computación, se prioriza tomar dicho conocimiento y adaptarlo a un flujo que pueda servir para el caso de la gestión del contenido del *CdC* de *LeanSight*. En particular toma como base la metodología de versionamiento *Git*, descrita en la sección 2.1.2.

Una segunda alternativa es diseñar desde cero un flujo de versionamiento, aunque esto no se considera óptimo, dado que ya existe un trabajo considerable, de varios años de maduración de los sistemas de versionamiento, por lo que se desecha esta alternativa.

4.4.4. Diseño de la solución

Se toma el sistema de versionamiento de *Git* y se busca entender su funcionamiento a alto nivel, de tal manera de luego poder adaptarlo al caso de la empresa. Para ello se parte desde la situación actual, planteada en la sección 3.1. Además, se considera el uso de las herramientas y/o soluciones descritas durante este capítulo.

4.4.5. Implementación

La implementación que sigue están basada en el uso de las distintas partes planteadas previamente, además del conocimiento del funcionamiento de la herramienta *Git* e *Issue Tracker*, adaptadas a la situación actual de la empresa.

A grandes rasgos, el proceso de cambio posee las siguientes etapas:

1. *Observación*: corresponde a identificar un problema, o síntoma.
2. *Caracterización del problema*: se describe el contexto del problema, es decir, situar dónde y cómo ocurre. Esto debiese quedar anotado en un lugar de acceso común a todos los diseñadores educacionales. Nótese que hay dos tipos posibles de modificación:
 - *Issue* (o problema): Corresponde a un problema que no está del todo especificado. Se transforma en cero o más cambios al haber consenso entre todos los diseñadores educacionales. Un ejemplo de esto sería “¿Es necesario agregar la actividad X en el módulo Y?”.
 - *Cambio*: Corresponde a una modificación en particular, por ejemplo cambiar una imagen dentro de una slide de la presentación.
3. *Dar inicio al cambio*: Se comprometen uno o más cambios, se parte desde el último punto estable. Un punto estable es aquél en que el contenido se considera apto para realizar una sesión. Entre los criterios considerados, está el hecho de haber o no utilizado el contenido en alguna sesión previa, además de considerar la opinión experta de los colaboradores.
4. *Modificación de los contenidos*: Se modifica algún soporte.

5. *Verificación de la propagación del cambio*: Se verifica si la modificación previa tiene algún impacto en otro soporte. De ser el caso, se debe volver al punto 4 para lograr coherencia entre los soportes.
6. *Dar fin al cambio*: se declara este como el último punto estable. Es posible volver al punto 3.
7. *Publicación*: Se publican los soportes en los formatos estándar para realizar los talleres (.pdf o .pptx). Este paso sólo debiese tener lugar en caso de necesidad, es decir tener que realizar un taller. La publicación se realiza automáticamente.
8. *Realización de taller*: Se dictan los talleres a un grupo de aprendices. Es posible que los facilitadores realicen una nueva observación y se vuelva a 1.
9. *Vivencia del taller*: los aprendices viven el taller y luego dan retroalimentación sobre su experiencia. Esta retroalimentación podría ser la base para volver al punto 1.

En la figura 4.12 se muestra un diagrama de flujo del proceso del cambio a alto nivel recién presentado.

4.4.6. Validación

Retroalimentación de los usuarios

Si bien el proceso presentado consiste en un proceso a alto nivel, aún no es posible definir con claridad de qué se trata un “Estado estable”, y por ende tampoco es posible definir un “Estado inestable”. Esto se debe principalmente a que incluso los módulos que poseen la menor cantidad de cambios, son modificados regularmente. A raíz de esto, surgen las siguientes preguntas de los usuarios:

1. ¿Qué es un estado estable? ¿Qué es un estado inestable?
2. ¿Cómo es posible indicar o denotar la estabilidad/inestabilidad de un módulo o sesión?
3. ¿Bajo qué criterio se hace la distinción entre ambos estados? ¿De quién o qué depende esto?
4. Dado que, según el proceso presentado, solo un módulo o sesión estable puede ser publicado, ¿de qué manera se pueden relacionar ambas cosas para obtener más información sobre un estado estable/inestable?

Por otro lado, el hecho de que el proceso esté basado en parte en las herramientas anteriores, hace que su validación sea dependiente de la validación de las otras partes. En otras palabras, si la implementación del resto de la solución es validada, entonces el proceso de cambio a alto nivel también.

Modificaciones realizadas a partir de la retroalimentación

A partir de las preguntas de la sección anterior, fue posible realizar las siguientes modificaciones:

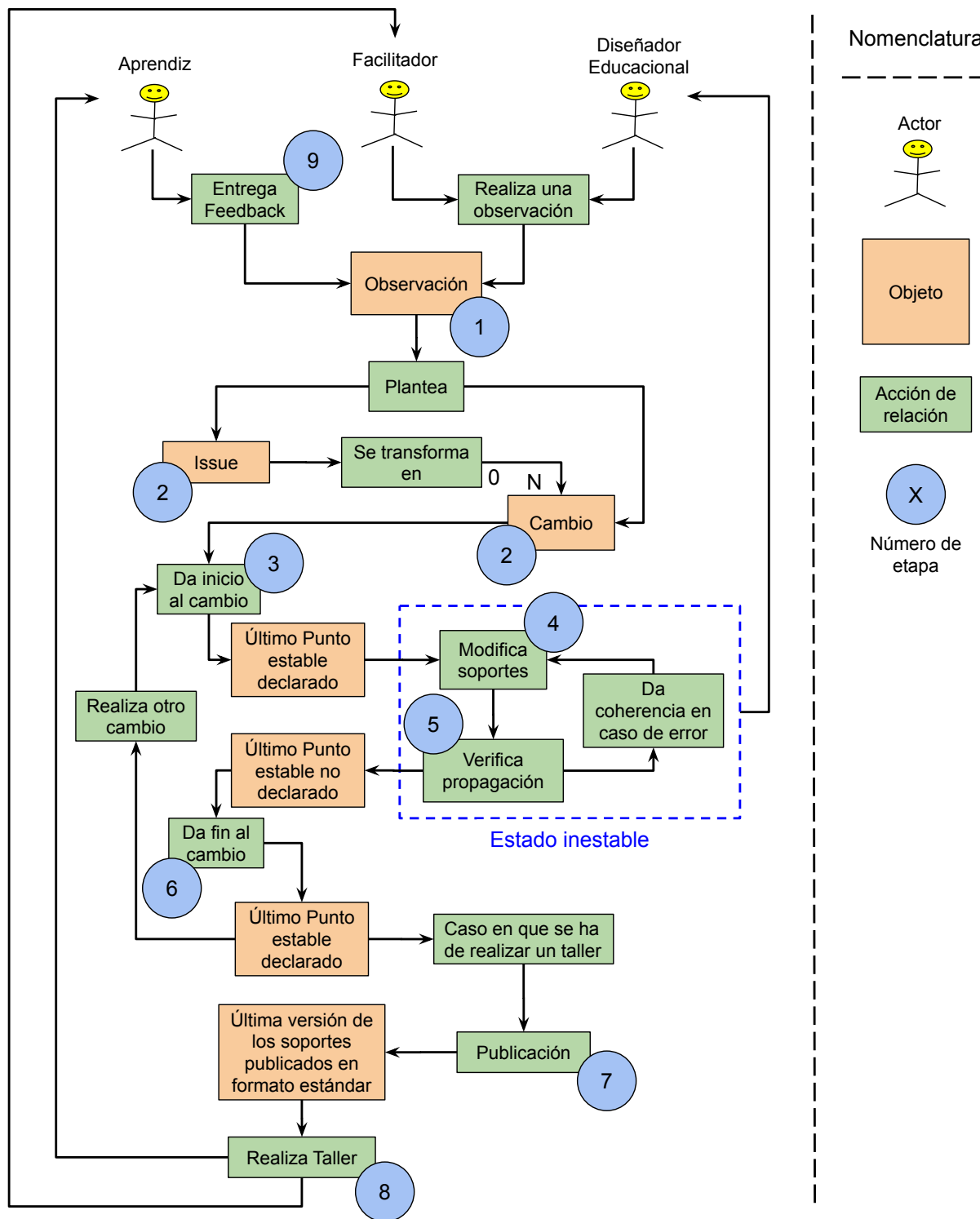


Figura 4.12: Diagrama de flujo del Proceso de Cambio a alto nivel

Con respecto a las preguntas 1 y 4, aún no es posible determinar qué denota cada estado, pero si se sabe que está relacionado con la capacidad de publicar y posteriormente utilizar una sesión con un público determinado. A partir de esto, es posible determinar que la persona que mejor podría distinguir entre ambos estados es el facilitador, quién está a cargo de la publicación y realización de los talleres, por lo que a partir de su experiencia es capaz de intuir si un módulo se encuentra estable o no. Esto responde a la pregunta 3.

Queda pendiente poder responder a la pregunta 4.

4.4.7. Aprendizajes

Ahora que el proceso de cambio está definido, es posible identificar los subprocesos que pueden ser automatizados, tales como la publicación o creación de sesiones. Además, se hace necesario considerar una manera ordenada de acceder a los contenidos del *CdC* que permita aplicar el flujo de cambio propuesto.

4.5. *Navegador*

4.5.1. Situación actual

A partir de los incrementos de solución previamente descritos, es posible:

- Caracterizar los incidentes, correspondiente a issues y cambios asociados.
- Conocer el ciclo de vida, o flujo, por el que pasan los cambios e issues.
- Visibilizar y dar acceso a todo el equipo a la información pertinente correspondiente a los cambios e issues a través de *Issue Tracker*.
- Visibilizar el relato lógico de los módulos a través del Modelo de Módulos Educativos o *MoME*, el cual está conectado con el *Issue Tracker*, de tal manera de poder asociar incidentes a elementos específicos de un módulo.

No obstante, si bien es posible realizar modificaciones del contenido del *CdC*, aún no es posible acceder de manera ordenada a las distintas versiones de los módulos o sesiones, correspondiente al objetivo específico 3.

En otras palabras, se quiere atacar los siguientes criterios de aceptación: F-3, F-4, F-5, F-6, F-7, F-8, F-9, F-10, F-11 y F-12.

Actualmente, los facilitadores aún siguen preparando las sesiones a través del uso del *CdC*, lo que implica navegar hacia cada carpeta representando un módulo o sesión, basado en el conocimiento o experticia que estos poseen. Sin embargo, en el caso de aquellos facilitadores que poseen menos conocimientos sobre los contenidos del *CdC*, les es difícil acceder la información de manera directa. Por otro lado, acciones como las de publicar los contenidos en

preparación de la realización de una sesión, aún sigue siendo un proceso largo que requiere de un mínimo de decisiones, pese a necesitar muchas acciones que podrían ser automatizadas.

4.5.2. Objetivo del ciclo

Este ciclo tiene por objetivo identificar la mejor manera de acceder o navegar los contenidos del *CdC* de manera ordenada y lógica por todo el equipo, habilitando la creación de sesiones a partir de los contenidos de sesiones anteriores o módulos. Además, se busca facilitar la publicación de los contenidos de una sesión previo a su realización.

4.5.3. Alternativas de solución

La solución que existe actualmente para navegar los contenidos es acceder directamente al directorio de *Google Drive* de la empresa, lo que necesita de ciertos conocimientos para poder ubicar dónde se encuentra el contenido. Es por ello que lo ideal sería tener algún soporte que permitiese navegar los contenidos de manera simple y ordenada. Además, debe poseer la capacidad de interactuar con el usuario para poder crear sesiones a partir de los contenidos navegados. En otras palabras se hace necesaria una interfaz de usuario que sea capaz de acceder los contenidos del *CdC* para mostrarlos de manera simple y ordenada a los usuarios. Para desarrollar tal interfaz, se proponen dos soluciones: una aplicación que se instale de manera local dentro de los computadores de los usuarios, o una aplicación web.

La aplicación local tiene la ventaja de no depender de un lenguaje de programación específica, por lo que da mayor libertad a la hora de integrar distintas herramientas en función de las necesidades de la solución. No obstante, posee la desventaja de tener que ser descargada y actualizada en caso de cambio, además de depender tanto del sistema operativo como de la plataforma desde la cual se utiliza (PC o Smartphone). Considerando que existen colaboradores que utilizan OSX y otros Windows, esta solución parece no ser simple de implementar.

En lo que respecta a la aplicación web, posee la gran ventaja de no depender de la plataforma, además de ser utilizable tanto desde un computador como un smartphone, lo que le da mayor flexibilidad a la hora de ser utilizada. La desventaja es que depende de lenguajes específicos, tales como *HTML* y *Javascript* o *Typescript*.

Dada la gran flexibilidad de la aplicación web, se decide perseguir esa solución.

4.5.4. Diseño de la solución

La aplicación web debe poseer las siguientes características:

1. Acceder al contenido del *CdC*.
2. Ser accesible por todo el equipo, y no permitir el acceso a personas externas.

3. Mostrar tanto los módulos como las sesiones realizadas con anterioridad.
4. Mostrar el contenido de forma simplificada al usuario.
5. Permitir distinguir entre los módulos y sesiones, de manera simple y ordenada.
6. Dar acceso a todos los contenidos de tal manera de poder compararlos entre sí.
7. Permitir la creación de sesiones a partir del contenido, es decir tanto los módulos como las sesiones anteriores.
8. Permitir publicar el contenido de una sesión de manera automática, previo a la realización de esta.

Con las características listadas, es posible explorar alternativas de solución.

4.5.5. Alternativas de implementación

Las alternativas de implementación se basan en las características listadas en la sección anterior, por lo que se procede a listar alternativas que permitan cumplir con dichas características, para llegar a la más adecuada.

Con respecto a la característica 1, dado que el *CdC* se encuentra en *Google Drive*, es posible utilizar tanto la *GAS API* como *GAS* para accederlo. La primera es accesible por cualquier aplicación con acceso a internet, mientras que la segunda está integrada dentro del ambiente de desarrollo de *Google G Suite*.

Con respecto a la característica 2, es necesario que solo el equipo pueda acceder al contenido, y considerando que el dominio de la empresa fue creado con una cuenta profesional de *Gmail*, es directo poder identificar a los colaboradores al utilizar el ambiente de *Google*.

Con respecto a la característica 3, los módulos y sesiones anteriores se encuentran en *Drive*, por lo que al acceder a este, es posible listarlos. No obstante, es necesario diferenciar entre lo que son los módulos, las sesiones y otros contenidos. Es por ello que se hace necesario identificar estos elementos. Para ello se propone listarlos, para luego ver de qué manera recorrerlos. Una opción es utilizar archivos de metadatos que denoten la existencia de un módulo o sesión, o simplemente apuntar directo al archivo a través del identificador único que posee cada archivo de *Drive*.

Con respecto a la característica 4, para poder mostrar el contenido de forma simplificada, es necesario identificar qué elementos son necesarios para poder caracterizar un módulo o sesión. Para ello, es posible tener un nombre que sea distintivo y único, además de poseer información sobre los contenidos presentes en cada contenido. Esto se puede lograr al listar los contenidos bajo cada nombre, con algún tipo de estructura visual que permita diferenciarlos, tal como tamaños de letra distintas, o colores.

Con respecto a las características 5 y 6, de manera similar al punto anterior, distinguir entre distintos elementos se puede hacer a través de un nombre distintivo, o al tener la capacidad de acceder al contenido de manera directa, por ejemplo a través de un hipervínculo que lleve a los contenidos de un módulo o sesión.

Con respecto a la característica 7, la interfaz debe ser capaz de interactuar con el contenido del *CdC*, de tal manera de poder crear nuevas sesiones a partir de este contenido. Para ello es necesario diseñar una estructura base que represente un módulo y una sesión, la cual puede ser reproducida de manera programática. Además, es necesario poseer una arquitectura que permita acceder a los datos del *CdC*.

Con respecto a la característica 8, la interfaz debe también permitir la publicación de manera programática de los contenidos de una sesión, para ello, utilizar *GAS* permite acceder a todos los productos de *Google*, por lo que se hace posible publicar.

A modo de resumen, las ventajas que posee el ambiente de desarrollo integrado de *Google G Suite* lo hacen el candidato ideal para implementar la solución del *Navegador* de contenido del *CdC*.

4.5.6. Implementación

Organización del trabajo de implementación

Es importante notar que el trabajo de implementación realizado para poder navegar los módulos y sesiones fue hecho en colaboración con Franco Cruces, quién es parte del equipo de desarrollo de la empresa. La división del trabajo se realizó en base a pequeñas tareas asociadas a funcionalidades específicas, repartidas de manera equitativa entre ambos desarrolladores, en función de la disponibilidad de tiempo y afinidad con las tareas realizadas. En varias oportunidades se realizó programación de a pares, la cual consiste en programar en pareja, con un solo computador, lo que permite paralelizar el trabajo, además de revisar el código a medida que se escribe. El trabajo de desarrollo fue realizado en un 40% por Franco y el 60% restante por el autor de este documento, en particular, la implementación de todo lo relacionado con el estado de progreso; gran parte del “look&feel”; implementación de los botones de “Publicar” y escalamiento de otras funcionalidades.

Trabajo realizado

La implementación consiste en una aplicación web realizada utilizando *GAS*, de tal manera de aprovechar el ambiente de desarrollo de *Google G Suite*, el cual permite utilizar los productos de *Google*, en particular *Drive* en donde se encuentra el *CdC*, además de *Slides*, *Spreadsheet* y *Docs*, donde están implementadas las presentaciones, el *MoME* y los cuader-nillos. Además, la identificación de los miembros del equipo se realiza utilizando los mismos mails del dominio de la empresa, dado que estos utilizan *Gmail*.

La idea del *Navegador*, consiste en mostrar todas las sesiones y módulos existentes, poder acceder a dichos contenidos a partir de un hipervínculo, y a su vez poder seleccionar los distintos contenidos para crear sesiones nuevas y posteriormente publicarlas.

Lo primero es caracterizar lo que es un módulo o una sesión, para ello se utilizan archivos

```

{"title":"Módulo - Planificación Incremental",
"files":[
  {"displayName":"Checklist - Planificación Incremental ",
    "id":"1Shz003APrjS7sXEfSxEhET4Ejtqh1JQEjNgcNnJbzi0",
    "type":"checklist",
    "mime":"application/vnd.google-apps.document",
    "origin":{"fileId":"13kSEphbnIUQGjMYZ62YQPfrMt907tquMBZM4xteYnfU"}
  },
  {"displayName":"Módulo - Planificación Incremental - Presentación",
    "id":"1efGL3YIOXeLwxiW4BboepC8dvagqtvLDWVD2QtPyuig",
    "type":"slides",
    "mime":"application/vnd.google-apps.presentation",
    "origin":{"fileId":"136TNmw8wGQghkd3XXbX2Z01EzBOLZft2pmj0zCWAs1g"}
  },
  {"displayName":"MoMe - Planificación Incremental",
    "id":"1LwkSCM9IJBmYYARmar6N3iN9Asd4FQM73Ejn7AojGU",
    "type":"mome","mime":"application/vnd.google-apps.spreadsheet",
    "origin":{"fileId":"1S3WIFxEBdP4HH3emCBSeIKFT039dMVRkLV1kecXdMzQ"}
  },
  {"displayName":"Módulo - Planificación Incremental - Guía del Alumno",
    "id":"1lhBuaheXD2LAY2N15oI-fm14AlDRcmkmqnIkZLm6Xdc",
    "type":"workbook","mime":"application/vnd.google-apps.document",
    "origin":{"fileId":"1ESQMr4CBEJV0aXyYtRfjS4IGy2WwQY-CmoAR28DDcPI"}
  }
]
,"origin":{"moduleId":"1Ap6xA1jHIhV-5XAlEDMVVwv_EffIX1A1"}}

```

Figura 4.13: Captura de pantalla del contenido de un archivo de metadatos para el caso de un módulo.

de metadato. En el caso de un módulo, el de metadato contiene los siguientes elementos:

- Título del módulo
- Una lista de archivos: esta posee los distintos contenidos del módulo, los cuales contienen la siguiente información:
 - Nombre del archivo
 - Identificador único del archivo
 - Tipo de archivo: si es que se trata de un cuadernillo, presentación, *MoME*, Checklist, u otra cosa.
 - Mime: tipo de documento según el formato de *Drive*.
 - Origen: es decir el identificador del archivo de origen.
- Origen: es decir el identificador del archivo de origen, en caso de ser una copia. Esto permite volver al origen del archivo en caso de ser necesario.

En la figura 4.13, es posible ver una captura de pantalla del contenido de un archivo de metadato de un módulo.

En el caso de una sesión, el archivo de metadatos consta de los siguientes elementos:

```

{"moduleIds":
  ["1AasjY4lKrKqTvj5W3aR70Tyv8-VwKC5K",
   "1d5W0aTdE6cqNCZawUnGic5zZxnwwk7jM",
   "1Hryxw7QpUOCP8MddUIgmq4WOTDajyj0l"],
 "title":"20190624 - Bci Fundamentos"}

```

Figura 4.14: Captura de pantalla del contenido de un archivo de metadatos para el caso de una sesión.

- Una lista de módulos: esta contiene los identificadores únicos asociados a los archivos de metadato de los módulos que contiene la sesión.

En la figura 4.14 es posible ver una captura de pantalla del contenido de un archivo de metadato de una sesión.

Luego es necesario poder acceder de manera simple a la lista de sesiones y módulos, para ello se crea una lista conteniendo una lista de los módulos y una lista de las sesiones. En el caso de las sesiones, la información está contenido en un archivo de nombre “.ls_session_list”, la cual contiene una lista de sesiones, representadas por una fecha de creación y un identificador asociado al archivo de metadato de dicha sesión, una captura de pantalla del contenido está presente en la figura 4.15. En el caso de los módulos, el archivo de metadatos, de nombre “.ls_oficial_modules” contiene una lista de identificadores de los archivos de metadatos asociados a los módulos. En la figura 4.16 es posible ver una captura de pantalla.

```

{"sessions":[
  {"created_on":"2019-06-06T15:09:13.459Z","id":"11iSFBasQeHFDGUolplwfw_198G80PKE_"},
  {"created_on":"2019-06-06T15:29:17.908Z","id":"1JVti6yY6b7N5p-52Xpflx16QeLtjuLee"},
  {"created_on":"2019-06-06T17:13:12.950Z","id":"1nA-dXr4e8YnT2EecN7TXjsxCBmmNFYZv"},
  {"created_on":"2019-06-06T17:14:37.207Z","id":"1ZIGR0Tfs2bMKHTwir3GIIdUIsNHuyzF56"},
  {"created_on":"2019-06-06T17:27:11.137Z","id":"1R4AGDfsJft7k1hZlDUCzIAMU11JwDT_J"},
  {"created_on":"2019-06-06T17:41:18.105Z","id":"1AOTbz1z3NmfHto4wZp25z7qjaGXA0tLa"},
  {"created_on":"2019-06-06T17:42:39.286Z","id":"1ieBrWAQ8aZ71258LU3qtsu5J0r5Pkvtx"}
]
}

```

Figura 4.15: Captura de pantalla del contenido de un archivo de metadatos conteniendo la lista de sesiones.

Acto seguido es necesario poder mostrar todo el contenido en una interfaz web. Para ello se utiliza *GAS*, producto que permite implementaciones web. Es necesario utilizar los lenguajes *HTML* y *Javascript* para ello. Lo interesante de esta herramienta, es que cada producto de *Google* actúa como una aplicación, que posee una API en particular, y que es accesible. En otras palabras, todo el contenido presente en productos de *Google* es accesible, en particular los elementos que componen un módulo, además de archivos binarios como son los archivos de metadatos.

Lo siguiente es poder acceder a todo el contenido desde la interfaz. Como vimos en el párrafo anterior, es posible acceder a todos los elementos del *CdC* que componen un módulo o sesión, además de poseer listas que contienen la totalidad de módulos y sesiones existentes.

```

{"moduleIds":
  [
    "1uDN4Aan33N6IMVH0Z8sIp2hA1ltBDFpx",
    "1xdHFoqWAZqyyS8xSq7swNvs4Ftqu7Ema",
    "1_jYerdCPmayxw5j2Z1LTZ91loG-GTw9Z",
    "10Bt1QvaZc1t-aoCWLhwcVEIHUJsF3_wh",
    "1hhmcnIURI__3kudR0rR7gVYW7Hu2PP-7",
    "1bQToiwsx0HahejYk_n6ZUGTMwvjDsZNs",
    "1NHTCN7kc_AT1qUyR_7GzKXHGydhEvFTG"
  ]
}

```

Figura 4.16: Captura de pantalla del contenido de un archivo de metadatos conteniendo la lista de módulos.

En otras palabras, es posible interactuar con dichos elementos, la pregunta es cómo. Para ello se separa el contenido de la interfaz en dos partes principales, la primera consiste en la parte superior, que consiste en una entrada de texto para poder nombrar una sesión, acompañada de un botón “Crear” que permite crear los módulos. La segunda parte está separada en dos columnas, que dividen la pantalla en dos: a la izquierda se encuentra la lista de sesiones, a la derecha la lista de módulos. La lista de las sesiones posee todos los módulos que estas contienen, además de su nombre y fecha de creación. En lo que respecta a la lista de módulos, estos poseen sus nombres respectivos además de contenido asociado. Cada módulo posee un checkbox al lado de su nombre, esto es cierto tanto para los módulos contenidos en una sesión como en un módulo. Además, cada nombre es a su vez un hipervínculo que lleva directamente a la carpeta de módulo o sesión, esto incluye los contenidos de cada módulo, es decir cuadernillo, presentación, *MoME* y Checklist.

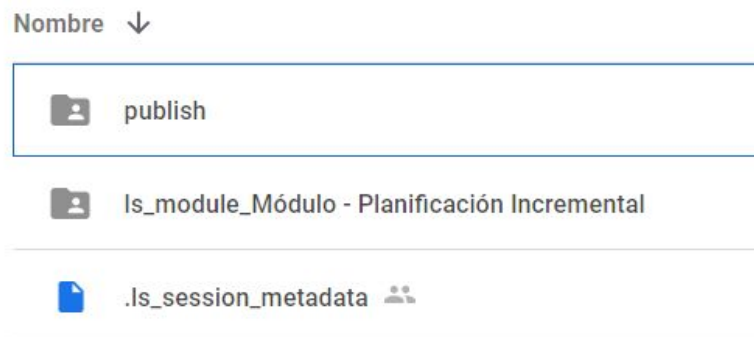
Llegado a este punto, es necesario poder crear sesiones nuevas, y poder distinguirlas entre sí, esto es posible a través del uso de nombres de sesiones que denoten la fecha de uso en formato *AAAAMMDD*, seguido del nombre del cliente. En lo que respecta al contenido de cada sesión, los módulos están listados bajo el nombre de la sesión, y cada sesión se encuentra separada por un espacio, el cual permite separar visualmente los elementos.

Finalmente, es necesario tener la posibilidad de publicar los módulos desde la misma interfaz. Para ello, se agrega la opción de poder publicar a partir de un botón con el texto “Publicar” a la derecha del nombre de cada sesión o módulo. Esto realiza una publicación dentro de la carpeta de la sesión o módulo, en un formato adecuado para realizar una sesión, es decir .pdf para el cuadernillo y la presentación, y .pptx para la presentación también. Un ejemplo de publicación está presente en la figura 4.17.

Una vez implementadas todas estas funcionalidades, es necesario dar al usuario retroalimentación del estado del proceso de publicación, o creación de una sesión, para ello se implementa un cuadro de texto que indica el estado del progreso de cualquiera de los dos procesos. Este recuadro se encuentra en la parte superior derecha de la interfaz.

La interfaz resultante de lo anteriormente descrito se encuentra en la figura 4.18.

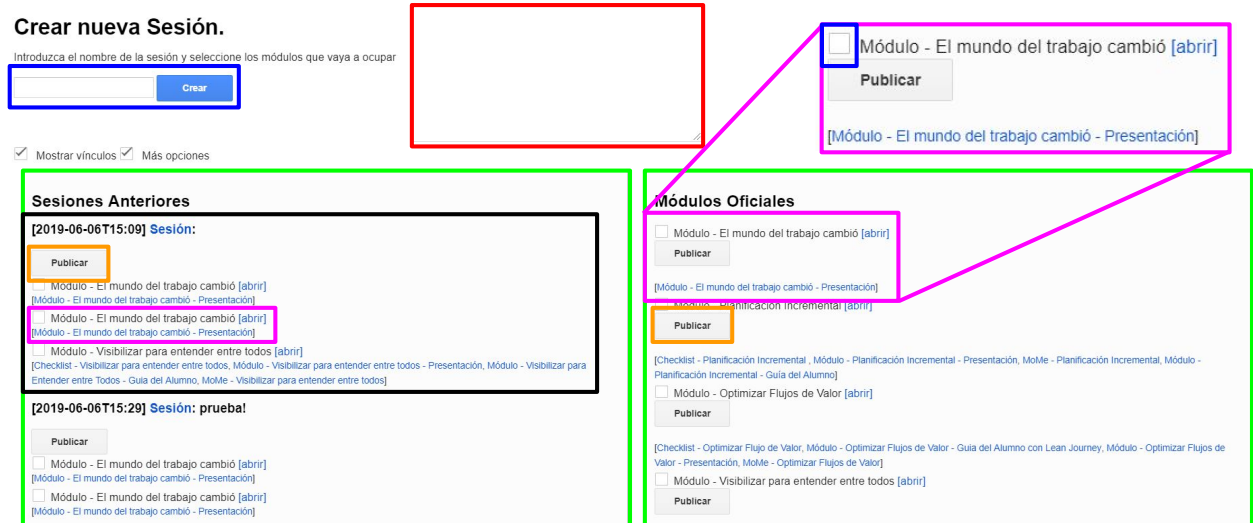
Contenido de una carpeta de sesión



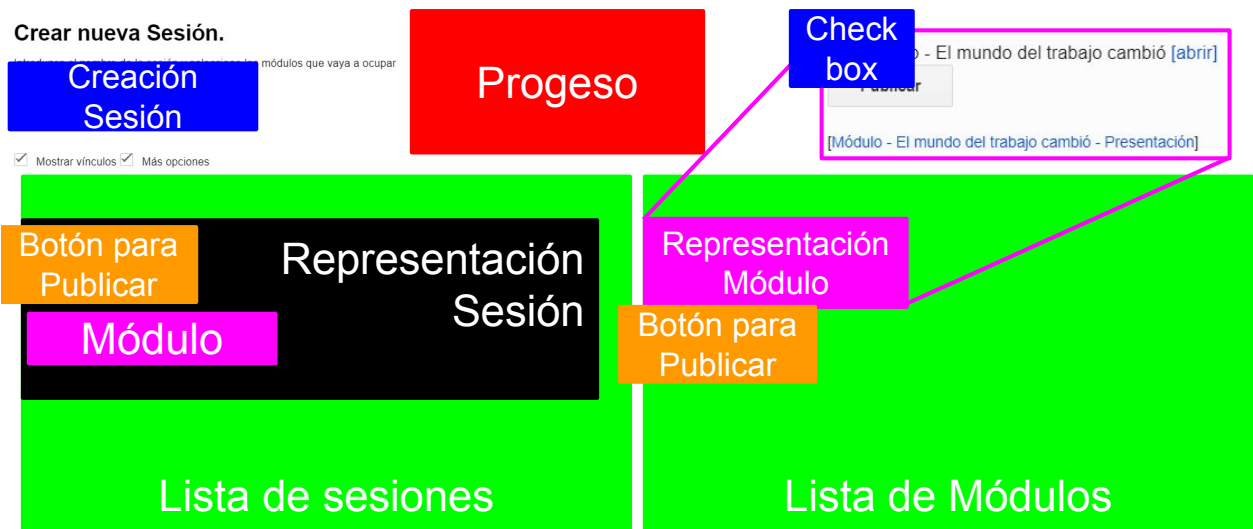
Contenido de la carpeta publish dentro de una sesión



Figura 4.17: Ejemplo de contenido de una carpeta de sesión, incluyendo la carpeta con el módulo, la carpeta “publish” con su contenido y el archivo de metadatos “.ls_session_metadata”.



(a)



(b)

Figura 4.18: Capturas de pantalla del *Navegador* inicial. En la figura 4.18(a) es posible ver una captura con los detalles del *Navegador*, mientras que en la figura 4.18(b) se ve el detalle de cómo se componen de manera abstracta.

4.5.7. Validación

Retroalimentación de los usuarios

Al utilizar el sistema, los usuarios plantearon las siguientes dudas:

1. La interfaz posee demasiada información ¿de qué manera es posible reducir la cantidad de información en pantalla, sin por lo mismo reducir las funcionalidades del *Navegador*?
2. Las sesiones más recientes están más actualizadas que las últimas, ¿cuál es la mejor manera de ordenar las sesiones?
3. El nombre de los archivos de metadato de las sesiones y módulos debiesen tener algún tipo de extensión clara que evite borrarlos, ¿qué nombre es adecuado para ello?
4. La creación de sesiones a partir del *Navegador* es parte de las funcionalidades necesarias.

Modificaciones realizadas a partir de la retroalimentación

A partir de la retroalimentación de los usuarios se realizan las siguientes modificaciones:

- Con respecto a 1, por defecto se esconden las opciones de publicación además de los contenidos de los módulos, lo que reduce la cantidad de información. No obstante, aún es posible acceder a dichas opciones, al poseer un checkbox para mostrar más opciones y/o mostrar los contenidos.
- Con respecto a 2, se invierte el orden en que se muestran las sesiones, por lo que la primera sesión a mostrarse es la última, lo cual es de mayor utilidad, dado que es más reciente.
- Con respecto a 3, los archivos de metadatos tenían como nombre “.metadata”, lo cual era poco descriptivo y cabía la posibilidad de borrarlos. Es por ello que se crea la extensión “.ls_module_metadata” y “.ls_session_metadata”, de tal manera de distinguir claramente de qué se trata.
- Con respecto a 4, se agrega la posibilidad de crear módulos nuevos a partir del *Navegador*, lo que crea una carpeta poseyendo todas las características de un módulo. No obstante, resta por implementar la posibilidad de agregar y quitar contenido, de tal manera de volver la herramienta del *Navegador* algo completo.

En la figura 4.19, es posible ver los cambios realizados.

4.5.8. Aprendizajes

Con la implementación del *Navegador*, se pone en manos de los usuarios una herramienta que permite apoyar el proceso de creación y gestión del contenido del *CdC*. Por otro lado, permite integrar el uso de las distintas herramientas desarrolladas durante el transcurso de este trabajo. Si bien aún queda mucho por mejorar para realmente hacer de esta herramienta algo utilizado por todos los usuarios, se está bien encaminado.

4.6. Resumen

Finalizado este capítulo, el lector es capaz de entender la lógica detrás de la construcción de cada una de las partes de la solución. En particular el versionamiento, para diferenciar entre los distintos elementos del *CdC*; el uso de un *Issue Tracker* adaptado a la empresa, que se utiliza en conjunto con el *MoME*, de tal manera de poder identificar y ubicar la necesidad de cambio dentro de los contenidos; la definición de un flujo de cambio, que permite ligar tanto la modificación, con la aplicación de los contenidos en sesiones, las cuales luego generan retroalimentación que sirve para generar nuevamente modificaciones; la implementación del *Navegador* de contenidos, que facilita su acceso a los facilitadores, en particular al habilitar la publicación automática. Cabe resaltar que los criterios de aceptación y de calidad son aquellos que guían el diseño e implementación.

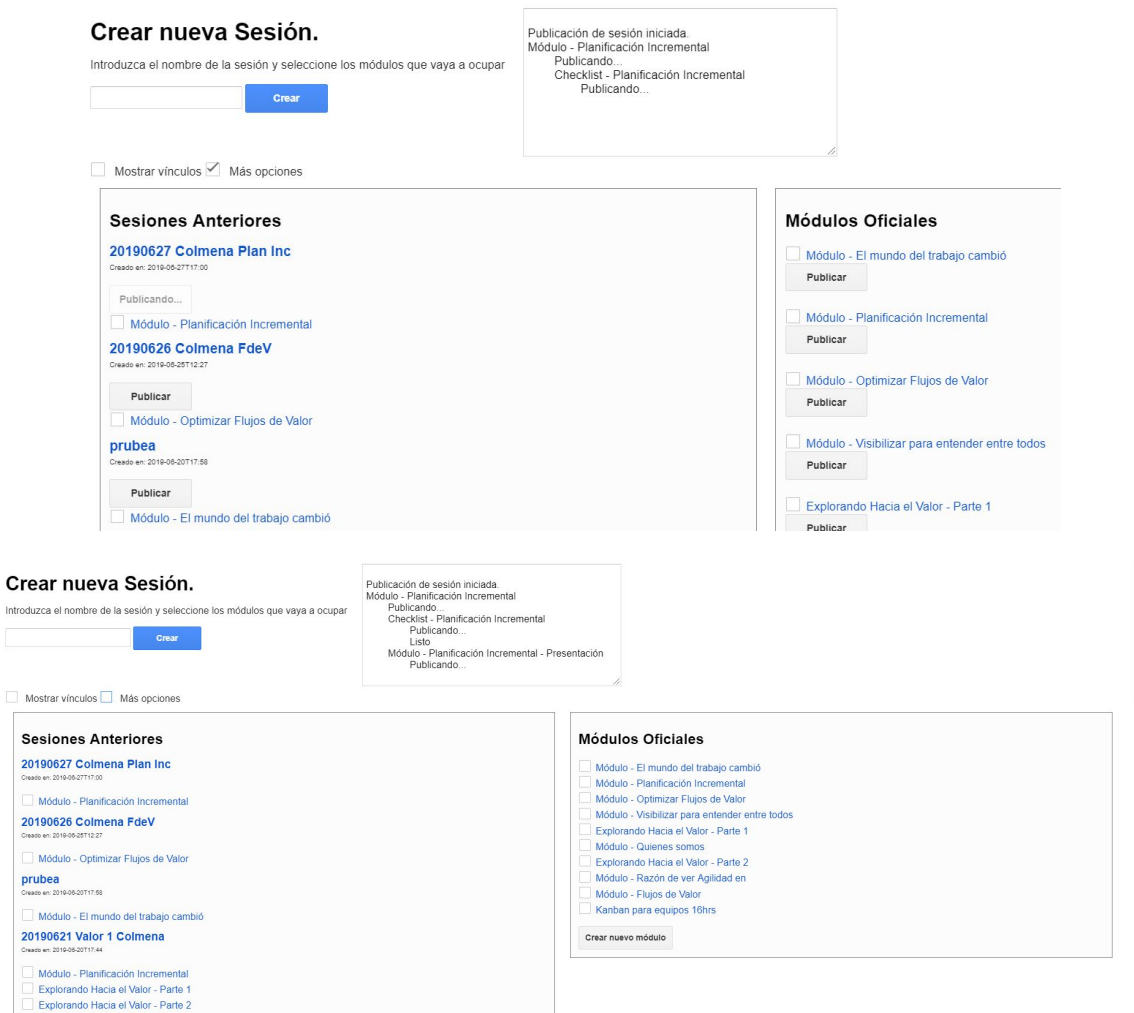


Figura 4.19: Captura de pantalla del *Navegador* luego de las modificaciones realizadas a partir de la retroalimentación por los usuarios. Nótese que la manera de distribuirse el contenido es muy similar a aquél presentado en la figura 4.18, la gran diferencia reside en que hay menor información en pantalla, por lo que el usuario posee mayor facilidad para navegar los contenidos.

Capítulo 5

Validaciones

En el caso de este proyecto, la descripción de la solución en el capítulo 4 posee una validación basada en retroalimentación de los usuarios del sistema. A modo de recordatorio, en la sección 5.1 se describirán las principales características de dicha validación, dando énfasis a la relación de estas con los criterios de aceptación respectivos.

Acto seguido se describe el diseño y realización de dos estudios de usabilidad: ambos consisten en una evaluación de usabilidad, el primero para un facilitador, presente en la sección 5.2; el segundo para un diseñador educacional, presente en la sección 5.3. Para ambos estudios, se plantea el objetivo, los participantes, la metodología y los materiales necesarios, para finalmente describir las tareas que los participantes deben poder realizar. Cada estudio arroja resultados, los cuales están asociados a determinados criterios de aceptación. A partir de esto, es posible medir la usabilidad y utilidad de la solución con respecto al comportamiento deseado.

Por otro lado, se busca realizar un análisis cualitativo de la percepción de los usuarios frente a la solución propuesta. De manera concreta, se quiere determinar si los usuarios consideran que la solución es un aporte para la empresa. Para ello se utiliza una encuesta cualitativa basada en la Escala de Usabilidad de un Sistema (EUS), o *SUS* por sus siglas en inglés. Este sistema se explica en la sección 2.3.7.

5.1. Resumen de las validaciones intermedias

A continuación se presenta un resumen de las validaciones intermedias, realizadas durante el desarrollo de la herramienta y en cada ciclo. El objetivo es resumir brevemente las principales validaciones y de qué manera impactaron el desarrollo de la solución propuesta.

En lo que respecta la caracterización de los cambios explicada en la sección 4.1, es decir, versionar los distintos módulos y sesiones, fue posible determinar una nomenclatura que posee una granularidad adaptada al ciclo de vida de creación de estos últimos. Además, se logró encontrar una nomenclatura que evitase confusiones con archivos temporales, y por ende

permitir a los usuarios identificar claramente los archivos versionados.

Para caracterizar el cambio y su flujo, explicado en la sección 4.2, la solución está basada en el funcionamiento de un *Issue Tracker*, implementado a través de la herramienta *Trello*. Además, se utilizó el flujo propuesto de *Discovery - Delivery* propuesto por *Optimizely* [29] para llevar al cabo el seguimiento de los distintos incidentes que puedan observar los diseñadores educativos. A raíz de la retroalimentación de los usuarios al usar esta herramienta, es posible afirmar que si bien se logró identificar la información pertinente para poder caracterizar los incidentes, la herramienta no es utilizada por los diseñadores educativos dada la falta de integración a su labor cotidiano, lo que significa trabajo extra. Por lo mismo, se hace necesario investigar de qué manera es posible integrar el *Issue Tracker* con otras herramientas desarrolladas, como el *Navegador* por ejemplo.

Al modelar los módulos educativos en la sección 4.3, se creó el *MoME*, el cuál, si bien poseía gran parte de la información necesaria para entender el relato lógico de un módulo, era de difícil lectura, al no permitir diferenciar los elementos entre sí. Para ello se utilizó una simbología o código de colores que permitiese contrastar los distintos tipos de elementos, además de denotar la importancia de cada uno de ellos. Por otro lado se cambió el vocabulario utilizado, para evitar confusiones, pasando de “Problema” a “Cambio” y de “Experimento” a “Issue”, lo cual permitía diferenciar los distintos tipos de incidentes de las distintas de un tipo de actividad pedagógica, además de utilizar términos más precisos y por ende afinar el vocabulario. Finalmente, se visibilizó la relación entre aprendizajes y paradigmas, siendo los paradigmas aquellos que justifican los distintos aprendizajes. Todo lo anterior fue modificado luego de obtener retroalimentación de los usuarios.

En lo que respecta el modelo del proceso de cambio a alto nivel explicado en la sección 4.4, este se basa principalmente en el funcionamiento de las herramientas del *MoME* e *Issue Tracker*, adaptadas al proceso de cambio de la empresa, además de permitir diseñar el funcionamiento del *Navegador*. Si bien no existen validaciones explícitas, implícitamente, el hecho de validar el resto de las herramientas entrega validez al flujo propuesto.

Finalmente, el navegador, presentado en la sección 4.5, la retroalimentación de los usuarios está enfocada en la interfaz usuario, buscando reducir la carga cognitiva al disminuir la cantidad de información presente, invirtiendo el orden de las sesiones dada la mayor pertinencia de sesiones más nuevas, además de evitar confusiones con los archivos de metadatos, al nombrarlos con una extensión que no fuese ambigua. Por otro lado, se determina necesario agregar la funcionalidad de creación y eliminación de módulos del navegador, de tal manera de poder utilizarlo como centro de control tanto por facilitadores como diseñadores educativos.

5.2. Estudio de usabilidad sobre el uso del *Navegador* por el usuario Facilitador

5.2.1. Diseño del estudio

Propósito del estudio

El objetivo de este test es validar la usabilidad de la herramienta del *Navegador* por parte del usuario Facilitador a través de un caso de uso que simula un caso real con usuarios reales, además de verificar si se cumplen los criterios de aceptación asociados a los distintos usuarios. Para ello se busca verificar si el facilitador es capaz de crear y posteriormente publicar una nueva sesión asociada a un cliente, fecha y contenidos específicos. Esto a partir del uso del *Navegador*, el cual contiene un número acotado de módulos y sesiones realizadas.

Cabe resaltar, que parte de las características deseadas de la sección 3.4.2, es tener una buena experiencia usuario, la cual se desglosa en abordabilidad por los usuarios, además de tanto simplicidad como usabilidad y testeabilidad de la solución. Por otro lado, el hecho de validar una herramienta que permita acceder y/o navegar a las distintas versiones de módulos y/o sesiones anteriores es el objetivo específico número 3. En otras palabras este estudio hace parte de los objetivos de este trabajo.

De manera concreta, el facilitador debe poder realizar una sesión al cliente *DCC*, fijada para el día 30 de junio del 2019 (190630) y que contenga los módulos:

1. “Visualizar para entender entre todos v2”
2. “El mundo del trabajo cambió”
3. “Explorando hacia el valor - parte 1”

Usuarios participantes

La herramienta del *Navegador* está diseñada para simplificar y acelerar el proceso de creación de sesiones y posterior publicación, roles asociados al usuario Facilitador. Para testear la herramienta, se prioriza tener usuarios con experiencia en la creación de sesiones, quienes estén familiarizados con el proceso anterior a la realización de este trabajo. De esta manera será posible evaluar la diferencia entre dicho proceso y el que se propone en este documento.

Materiales

Para probar que el facilitador sea capaz de utilizar el *Navegador*, se requiere de un computador con un navegador web y acceso a internet, además de la URL para acceder al *Navegador*. Éste debe poseer una cierta cantidad de elementos para poder llevar a cabo el test. Estos elementos se listan a continuación:

- El usuario sabe que anteriormente se realizó una sesión similar a la que necesita. Ésta fue realizada al cliente *DII*, y contiene los siguientes módulos:
 1. “Visualizar para entender entre todos v1”
 2. “El mundo del trabajo cambió”
- Existen los módulos oficiales:
 1. “Visualizar para entender entre todos v2”
 2. “Explorando hacia el valor - parte 1”
 3. “Explorando hacia el valor - parte 2”
 4. “El mundo del trabajo cambió”
- Existen las sesiones:
 - 190601 *DII*, con los módulos:
 1. “Visualizar para entender entre todos v1”
 2. “El mundo del trabajo cambió”
 - 190605 *DIMEC*, con los módulos:
 1. “Visualizar para entender entre todos v2”
 - 190620 *DII*, con los módulos:
 1. “Explorando hacia el valor - parte 1”
 2. “Explorando hacia el valor - parte 2”

Tareas a realizar

A continuación se plantean las distintas tareas. Cada cual posee un objetivo, un criterio para saber si se logró la tarea o no además de los criterios de aceptación asociados.

1. **Objetivo:** Identifico las sesiones realizadas al *DII*.
Se logra si el usuario identifica las sesiones “190601 *DII*” y “190620 *DII*”.
 Los CAs que se validan son: F-7, F-9.
2. **Objetivo:** De entre las sesiones realizadas al *DII*, identifico la sesión con los módulos “Visualizar para entender entre todos v1”, “El mundo del trabajo cambió”.
Se logra si el usuario identifica “190601 *DII*”.
 El CA que se valida es: F-8.
3. **Objetivo:** Verifico si la sesión “190601 *DII*” posee la última versión del módulo “Visualizar para entender entre todos” (v2)
Se logra si el usuario es capaz de acceder al módulo “Visualizar para entender entre todos v1” de la sesión “190601 *DII*”, y verifica la versión del módulo en el nombre de su versión publicada.
 El CA que se valida es: F-6.
4. **Objetivo:** Creo una sesión para el *DCC*, con fecha de realización de la sesión 190630, con los módulos “Visualizar para entender entre todos v2”, “El mundo del trabajo cambió”, “Explorando hacia el valor - parte 1”
Se logra si el usuario logra crear una sesión con el nombre “190630 *DCC*” con los módulos “Visualizar para entender entre todos v1”, “El mundo del trabajo cambió” y

“Explorando hacia el valor - parte 1”.

El CA que se valida es: F-12

5. **Objetivo:** Publico la nueva sesión creada y accedo al documento publicado automáticamente. **Se logra si** el usuario es capaz de publicar desde el *Navegador* la nueva sesión creada y luego accede a ella a partir del hipervínculo proporcionado.

Los CAs que se validan son: F-3, F-4, F-5.

Si se cumplen las tareas de “Identifico las sesiones realizadas al *DII*.” y “De entre las sesiones realizadas al *DII*, identifico la sesión con los módulos “Visualizar para entender entre todos v1”, “El mundo del trabajo cambió”.”, correspondientes a las tareas 1 y 2, entonces se cumple F-11.

Criterios de aceptación no considerados: F-1, F-2, F-10.

Este caso de uso consiste en preguntas guiadas dentro de un contexto ficticio que simula un caso de uso verdadero. Además considera 8 de los 11 criterios de aceptación relacionados con los facilitadores.

Metodología de estudio

La metodología de estudio utilizada es la de *Thinking Aloud* [16], la cual corresponde a pedir al usuario que verbalice sus pensamientos en voz alta a medida que utiliza el sistema para llevar a cabo las tareas requeridas. Esto permite entender de qué manera el usuario percibe e interactúa con el sistema, en particular ayuda a resaltar los posibles problemas de usabilidad al existir una diferencia entre la lógica del usuario y aquella del sistema.

Datos relevantes

Los datos relevantes que se busca obtener para su posterior análisis son los siguientes:

- Tiempo aproximado de realización de la tarea, en segundos.
- Número de tareas por realizar.
- Porcentaje de funciones relevantes utilizadas, en otras palabras qué criterios de aceptación se cumplen.
- Número de tareas completadas con éxito.
- Porcentaje de usuarios capaces de completar la tarea con éxito.
- Escala SUS [2] para medir la usabilidad de la herramienta. Para mayor información, ver la sección 2.3.7.

5.2.2. Ejecución del estudio

A continuación se presentan los resultados del estudio para cada uno de los usuarios entrevistados.

Resultados del usuario Usuario 1

Tareas	Tiempo (s)	Logrado (sí/no)	CA asociado
Tarea 1	20	Sí	F-7, F-9, (F-11)
Tarea 2	30	Sí	F-8, (F-11)
Tarea 3	- (*)	No	F-6
Tarea 4	30	Sí	F-12
Tarea 5	60	Sí	F-3, F-4, F-5

Tabla 5.1: Tabla de resultados del estudio de usabilidad del *Navegador* para el usuario facilitador, realizado por el facilitador Usuario 1. (*) El usuario no fue capaz de realizar esta tarea.

Comentarios del Usuario 1:

- Con respecto a la tarea 3, el usuario declara no saber cómo realizarla.
- Con respecto a la tarea 3, el nombre de la sesión/módulo podría contener la versión de dicho contenido.
- Con respecto a la tarea 3, se podría tener algo que indique dónde se encuentra la versión publicada del módulo o sesión.
- Con respecto a la tarea 5, al querer publicar el contenido de una sesión, el usuario marca los checkboxes asociados a los distintos elementos que desea publicar.
- Con respecto al funcionamiento de la herramienta, el usuario declara: “Esto está quedando muy bueno, está quedando práctico.”.

Resultados del usuario Usuario 2

Usuario 2	Tiempo (s)	Logrado (sí/no)	CA asociado
Tarea 1	15	Sí	F-7, F-9, (F-11)
Tarea 2	5	Si	F-8, (F-11)
Tarea 3	- (*)	No	F-6
Tarea 4	10	Si	F-12
Tarea 5	40	Sí	F-3, F-4, F-5

Tabla 5.2: Tabla de resultados del estudio de usabilidad del *Navegador* para el usuario facilitador, realizado por el facilitador Usuario 2. (*) El usuario no fue capaz de realizar esta tarea.

Comentarios del usuario 2:

- Con respecto a la tarea 3, no es explícito el lugar dónde está indicada la versión. Además, un módulo no debiese publicarse por sí solo, sino que dentro de una sesión, por lo que la manera actual de comparación no hace sentido. Se recomienda agregar la versión junto a la fecha de creación de cada uno de los módulos o sesiones.
- Con respecto al uso general del *Navegador*, el tiempo de carga de las sesiones es muy largo, y este crece de forma lineal con la cantidad de sesiones, por lo que solo va a

empeorar. Esto da una mala experiencia usuario, dado que es necesario esperar mucho tiempo antes de poder actuar. A modo de recomendación, se podría limitar la cantidad de sesiones que aparecen inicialmente a un número que se cargue dentro de un tiempo razonable, para luego dar la elección al usuario de querer cargar una mayor cantidad de sesiones.

5.2.3. Análisis de los resultados

En la tablas 5.3 y 5.4 se resumen los resultados de los estudios realizados por ambos usuarios, para luego entrar en el análisis de estos. En primer lugar se busca interpretar los resultados obtenidos, para luego entrar en detalles basados en los comentarios de los usuarios entrevistados.

Métrica	Usuario 1	Usuario 2
Tareas por realizar	5	
Tareas completadas con éxito	4	4
Éxito de tarea 1	100 %	
Tiempo tarea 1	20	15
Tiempo promedio tarea 1 (segundos)	17.5	
Éxito de tarea 2	100 %	
Tiempo tarea 2 (segundos)	30	5
Tiempo promedio tarea 2 (segundos)	17.5	
Éxito de tarea 3	0 %	
Tiempo tarea 3 (segundos)	-	-
Tiempo promedio tarea 3 (segundos)	-	
Éxito de tarea 4	100 %	
Tiempo tarea 4 (segundos)	30	10
Tiempo promedio tarea 4 (segundos)	20	
Éxito de tarea 5	100 %	
Tiempo tarea 5 (segundos)	60	40
Tiempo promedio tarea 5 (segundos)	50	
Tiempo total del estudio (segundos)	900	660

Tabla 5.3: Tabla recapitulativa de las tareas y los tiempos obtenidos en el estudio de usabilidad de la herramienta del *Navegador* por dos usuarios facilitadores.

Lo primero a notar en la tabla 5.3 es que el 80 % de las tareas por realizar fueron realizadas con éxito, en otras palabras solo una de las tareas no fue completada por ninguno de los usuarios. Para mayor detalle, las tareas 1, 2, 4 y 5 fueron logradas por ambos usuarios que participaron del estudio, mientras que ninguno logró la tarea 3. Cabe resaltar que si bien la tarea 3 posee una manera de ser resuelta, esta no es explícita y/o directa a partir del *Navegador*, dado que es necesario comparar versiones de elementos publicados dentro de cada sesión. Por otro lado, el hecho de poder comparar versiones es una funcionalidad necesaria para poder distinguir entre los contenidos de las distintas sesiones. En otras palabras, es

necesario agregar algún tipo de indicador de la versión, y tal como lo sugiere el usuario 2, sería posible agregarla al lado de la fecha de creación de la sesión, de tal manera de que sea visible.

En lo que respecta a los tiempos necesarios para la realización de las distintas tareas, es posible notar que los tiempos entre el primer y segundo usuario son distintos, de manera general, el usuario 2 realizó las tareas más rápido que el usuario 1, tomándole entre un sexto a un cuarto del tiempo. No obstante, la diferencia sigue siendo del orden de los 10 segundos, es decir del orden del 1-1.5% del tiempo total de la entrevista, lo cual no es muy significativo.

Otro resultado es que la tarea 5, correspondiente al hecho de publicar, toma en promedio 50 segundos, versus los en promedio 20 segundos que toman las otras tareas exitosas. A diferencia de las otras tareas que es posible realizar a partir de una interacción que estaba de manera aparente en la interfaz, la publicación necesita de un paso adicional para aparecer en la interfaz. Esto probablemente dificulta su uso, y por ende aumenta el tiempo necesario. Sería necesario evaluar si tal comportamiento es correcto, basado en cuán seguido se utiliza tal funcionalidad.

Por otro lado, el hecho de que las checkboxes se encuentren en el mismo contexto que el botón publicar lleva a los usuarios a pensar que ambas cosas están relacionadas, por lo que seleccionan los contenidos que quieren publicar, cuando ese no es el funcionamiento implementado. Es necesario evaluar si el botón publicar debiese encontrarse en un lugar distinto o si se podrían eventualmente desactivar las checkboxes al momento de activar la funcionalidad de publicación.

Métrica	Usuario 1	Usuario 2
Número total de CAs para los facilitadores	11	
Número de CAs evaluados	8	
Número de CAs validados	7	7
Porcentaje de CAs evaluados c/r al total	72 %	
Porcentaje de CAs validados c/r al total	64 %	64 %
Porcentaje CAs validados c/r a los evaluados	86 %	86 %

Tabla 5.4: Tabla recapitulativa de los porcentajes de CAs evaluados y validados en el estudio de usabilidad de la herramienta del *Navegador* por dos usuarios facilitadores.

Con respecto a la tabla 5.4, es posible afirmar que de los 11 CA relacionados con el facilitador, 8 fueron evaluados, lo que representa el 72% del total, lo que es cercano a un cuarto del total de criterios de aceptación. De la misma manera, 7 de los CAs fueron validados por el estudio, es decir, un 64% del total. Considerando solo los CAs evaluados, es posible afirmar que el 86% de los CAs evaluados fue validado con el estudio realizado. Cabe resaltar que este último porcentaje corresponde a los CAs correspondientes a tareas críticas, por lo que es un buen resultado.

Finalmente, se encuentra un error de implementación durante la realización del estudio: al dejar marcada la opción de “Más opciones” que permite activar la funcionalidad de publicación, en el caso de refrescarse los módulos o las sesiones, la funcionalidad de publicar desaparece de la interfaz aún cuando sigue marcada la opción de “Más opciones”. Esto indica

que es necesario arreglar la implementación para evitar el error.

5.2.4. Diseño de la evaluación utilizando la System Usability Scale (*SUS*)

La evaluación cualitativa utiliza el System Usability Scale (*SUS* a partir de ahora), también conocido como la Escala de Usabilidad de un Sistema (*EUS*). Tal y como se plantea en la sección 2.3.7, esta consiste en una serie de 10 aseveraciones, las cuales pueden ser evaluados con una escala que va del 1 al 5, dónde 1 denota que el usuario está “Totalmente en Desacuerdo” y 5 que el usuario está “Totalmente de Acuerdo”. A modo de recordatorio, las preguntas son las siguientes:

1. Creo que usaría este sistema frecuentemente
2. Encuentro este sistema innecesariamente complejo
3. Creo que el sistema fue fácil de usar
4. Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar este sistema
5. Las funciones de este sistema están bien integradas
6. Creo que el sistema es muy inconsistente
7. Imagino que la mayoría de la gente aprendería a usar este sistema en forma muy rápida
8. Encuentro que el sistema es muy difícil de usar
9. Me siento confiado al usar este sistema
10. Necesité aprender muchas cosas antes de ser capaz de usar este sistema

La razón por la que se utilizó esta escala, es para obtener una apreciación cualitativa rápida del uso de la herramienta, a partir de la cual poder evaluar el estado actual de la percepción de los usuarios con respecto a la herramienta utilizada. Uno de los beneficios de esta encuesta, es la posibilidad de poder repetirla sin costo en el futuro y observar si existe una diferencia en la apreciación de los usuarios.

Con respecto a los usuarios encuestados, se trata de los mismos usuarios que en el estudio anterior. Nótese que el estudio y la evaluación se hicieron uno tras otro, por lo que era interesante saber la percepción de los usuarios luego de verse enfrentados a utilizar la herramienta.

5.2.5. Ejecución de la evaluación

Para llevar a cabo la evaluación de usabilidad, se utiliza un *Google Forms* con todas las aseveraciones y la posibilidad de responder dada la escala del 1 al 5 presentada en la sección anterior. Este formulario estaba vinculado a un *Google Spreadsheet*, desde el cuál se calcula el índice de usabilidad. En las tablas 5.5 se encuentran los resultados de los dos usuarios entrevistados.

5.2.6. Análisis de los resultados

Aseveración	Puntaje usuario 1	Puntaje usuario 2
1. <i>Creo que usaría este sistema frecuentemente</i>	2	4
2. <i>Encuentro este sistema innecesariamente complejo</i>	4	3
3. <i>Creo que el sistema fue fácil de usar</i>	4	4
4. <i>Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar este sistema</i>	4	4
5. <i>Las funciones de este sistema están bien integradas</i>	4	4
6. <i>Creo que el sistema es muy inconsistente</i>	1	1
7. <i>Imagino que la mayoría de la gente aprendería a usar este sistema en forma rápida</i>	4	3
8. <i>Encuentro que el sistema es muy difícil de usar</i>	1	2
9. <i>Me siento confiado al usar el sistema</i>	4	5
10. <i>Necesité aprender muchas cosas antes de ser capaz de usar este sistema</i>	1	2
Puntuación SUS (sobre 100)	75	70
Puntuación promedio <i>SUS</i>	72.5	

Tabla 5.5: Tabla con los resultados de la encuesta cualitativa basada en SUS realizado a dos usuarios.

Los resultados de la evaluación se presentan en la tabla 5.5. De ellos es posible observar que la mayoría de las aseveraciones tiene puntajes similares, es decir con a lo más 1 punto de distancia, a excepción de las aseveraciones 1 y 3. Esto indica que la percepción de los usuarios es parecida a excepción de dichos puntos.

Al indagar en los resultados de las aseveraciones 1 y 3, se observa que el usuario 1 no cree que usará el sistema frecuentemente, dada una puntuación de 2 para la aseveración “Creo que usaría este sistema frecuentemente”, mientras que el usuario 2 piensa lo contrario, con un puntaje de 4. Además, es necesario considerar que el usuario 1 está en desacuerdo con la aseveración “Creo que el sistema fue fácil de usar” al poner un puntaje de 1, mientras que el usuario 2 puntuó con un 4 la misma aseveración. A partir de lo anterior es posible interpretar que el usuario 1 encuentra el sistema más complicado que el usuario 2. Esto se puede interpretar como una necesidad de mejorar el diseño de la herramienta, sobre todo basado en los comentarios y análisis expuestos en la sección 5.2.2 y posteriormente en la sección 5.2.3.

Por otro lado, la puntuación total en una escala de 100 es de 75 y 70 puntos respectivamente para el usuario 1 y 2, siendo un promedio de 72,5, lo cual está 4,5 puntos más arriba del puntaje asociado a un sistema con buena usabilidad, es decir 68 puntos, según los autores de la herramienta. En conclusión ambos usuarios estiman que el sistema tiene buena usabilidad,

aun cuando siempre hay espacio para mejoras.

5.3. Estudio de usabilidad sobre el uso del *MoME* y el *Issue Tracker* por el usuario Diseñador Educativo

5.3.1. Diseño del estudio

Propósito del estudio

El objetivo de este test es validar la usabilidad de la herramienta *Issue Tracker* en conjunto con el Modelos de Módulos Educativos (*MoME*) por parte del usuario Diseñador Educativo. Para ello, se simula un caso de uso real, basado en un cambio específico, con usuarios reales, además de verificar si se cumplen los criterios de aceptación asociados a los distintos usuarios. Para ello se busca verificar si el diseñador educativo es capaz de identificar un cambio a realizar, entender su razón de ser, realizar la modificación pertinente y finalmente denotar que dicho cambio fue realizado siguiendo el flujo propuesto en este documento.

Cabe resaltar, que parte de las características deseadas de la sección 3.4.2, es tener una buena experiencia usuario, la cual se desglosa en abordabilidad por todo usuario, además de tanto simplicidad como usabilidad y testeabilidad de la solución. Por otro lado, el hecho de validar un flujo estructurado de gestión de cambios para módulos de aprendizaje compuesto de múltiples documentos es el objetivo específico número 2. En otras palabras este estudio hace parte de los objetivos de este trabajo.

De manera concreta, el diseñador educativo debe ser capaz de identificar un cambio que no esté siendo realizado por otro diseñador educativo a partir del *MoME*, para luego entender la razón del cambio elegido a partir del *Issue Tracker*, realizar la modificación dentro de los soportes y denotar que esta fue completada dentro del *MoME*.

Usuarios participantes

Las herramientas del *MoME* e *Issue Tracker* están diseñadas para trabajar en conjunto y apoyar el proceso de cambio al proponer un flujo para realizar el proceso de cambio, el cual está asociado a los diseñadores educativos. Para testear el uso conjunto de la herramienta se prioriza tener usuarios que posean experiencia editando módulos y/o sesiones utilizando el proceso previo a la realización de este trabajo. De esta manera es posible evaluar la diferencia entre dicho proceso y aquel propuesto en este documento.

Materiales

El test corresponde al uso conjunto de las herramientas *MoME* e *Issue Tracker*, por lo que es necesario tener un computador con un navegador web y acceso a internet, además de la URL para acceder al Modelo del Módulo Educativo correspondiente al módulo “Visibilizar para entender entre todos”.

Este recurso (*MoME*) posee el relato lógico del módulo que modela, el cual posee un error específico, correspondiente a cambiar la imagen de una de las diapositivas. Dicha modificación se propaga al cuadernillo, por lo que es necesario realizar dos cambios. Además, posee el complemento *Trello for MoME* para Hojas de Cálculo (*Spreadsheet*) de Google desarrollado que permite automatizar el uso del *Issue Tracker* a partir del *MoME*.

Además, se utilizan dos pantallas de televisión de 40” conectadas al computador para poder visibilizar la totalidad de los recursos mientras se modifican.

Tareas a realizar

- Objetivo:** Identificar un cambio por realizar que no esté siendo realizado por otro diseñador educativo.
Se logra si a partir del *MoME*, el usuario es capaz de identificar los distintos cambios por realizar al módulo “Visibilizar para entender entre todos”, de los cuales uno se encuentra siendo realizado por otro diseñador educativo, y existe uno que debe ser realizado.
Los CAs que se validan son: DE-2, DE-7, DE-9, DE-10, DE-11.
- Objetivo:** Identificar la razón que sustenta el cambio dentro de los soportes.
Se logra si el usuario es capaz de entender que se debe cambiar la imagen de una diapositiva y de una página asociada, dado que cambió la línea gráfica (diseño) del módulo .
Los CAs que se validan son: DE-1, DE-5, DE-6.
- Objetivo:** Reconocer qué recursos deben ser modificados, y poder navegar hacia la diapositiva de la presentación y página del cuadernillo asociadas.
Se logra si el usuario es capaz de reconocer que el cambio de imagen corresponde a la diapositiva 20 y a la página 11 del cuadernillo del módulo “Visibilizar para entender entre todos”, para luego navegar hacia ambas dicha diapositiva/página.
Los CAs que se validan son: DE-3, DE-8, DE-15.
- Objetivo:** Modificar la imagen de la diapositiva de la presentación y la página del cuadernillo con la nueva línea gráfica.
Se logra si el usuario es capaz de describir los pasos para realizar un cambio de imagen en ambos soportes, acto seguido, debe volver al *MoME* y marcar el cambio como hecho. Solo se pide la descripción de los pasos a seguir en caso de querer cambiar una imagen, dado que es una actividad común de los diseñadores educativos, por lo que no se busca evaluar si son capaces de realizar, si no más bien si son capaces de seguir el flujo propuesto por este trabajo en conjunto con las herramientas desarrolladas durante el mismo.

Los CAs que se validan son: DE-14, DE-16.

5. **Objetivo:** Denotar dentro del MoME que el cambio fue realizado, y bajo qué circunstancias. **Se logra si** el usuario vuelve al MoME, denota que el cambio fue realizado al utilizar el componente *Trello-For-Mome* y utilizar la funcionalidad “Enviar a Done”.

Los CAs que se validan son: DE-12, DE-19, DE-21, DE-22, DE.23.

Los criterios que no fueron considerados son los siguientes: DE-4, DE-13, DE-17, DE-18, DE-20, DE-24, DE-25, DE-26, DE-27.

Este caso de uso consiste en preguntas guiadas dentro de un contexto ficticio que simula un caso de uso verdadero. Además considera 18 de los 27 criterios de aceptación relacionados con los diseñadores educacionales.

Metodología de estudio

La metodología de estudio utilizada es la de *Thinking Aloud* [16], la cual corresponde a pedir al usuario que verbalice sus pensamientos en voz alta a medida que utiliza el sistema para llevar a cabo las tareas requeridas. Esto permite entender de qué manera el usuario percibe e interactúa con el sistema, en particular ayuda a resaltar los posibles problemas de usabilidad al existir una diferencia entre la lógica del usuario y aquella del sistema.

Datos relevantes

Los datos relevantes que se busca obtener para su posterior análisis son los siguientes:

- Tiempo aproximado de realización de la tarea, en segundos.
- Número de tareas por realizar.
- Porcentaje de funciones relevantes utilizadas, en otras palabras qué criterios de aceptación se cumplen.
- Número de tareas completadas con éxito.
- Porcentaje de usuarios capaces de completar la tarea con éxito.
- Escala SUS [2] para medir la usabilidad de la herramienta. Para mayor información, ver la sección 2.3.7.

5.3.2. Ejecución del estudio

A continuación se presentan los resultados del estudio para cada uno de los usuarios entrevistados.

Usuario 1	Tiempo (s)	Logrado (sí/no)	CA asociado
<i>Tarea 1</i>	210	Sí, con dificultad	DE-2, DE-7, DE-9, DE-10, DE-11
<i>Tarea 2</i>	15	Sí	DE-1, DE-5, DE-6
<i>Tarea 3</i>	220	Sí, con dificultad	DE-3, DE-8, DE-15
<i>Tarea 4</i>	- (*)	Sí	DE-14, DE-16
<i>Tarea 5</i>	70	Sí	DE-12, DE-19, DE-21, DE-22, DE-23

Tabla 5.6: Tabla de resultados del estudio de usabilidad del *Issue Tracker* y el *MoME* para el usuario Diseñador Educacional, realizado por el usuario 1. (*) Ver tarea 4.

Resultados del usuario Usuario 1

Comentarios del Usuario 1:

- Con respecto a la tarea 1, el usuario se fija en el estado de la tarjeta para verificar si esta está siendo realizada o no. En otras palabras su conocimiento del flujo de un tablero Kanban ayuda.
- Con respecto a la nomenclatura para representar la celda del MoME que representa los contenidos por modificar, el usuario declara entenderla.
- Con respecto a la tarea 3, el usuario es capaz de identificar qué recursos/soportes deben ser modificados a través de las columnas que las representan dentro del *MoME*.
- Con respecto a la tarea 3, el usuario declara no saber cómo acceder el lugar específico dónde realizar el cambio a partir del *MoME*, sin embargo, es capaz de realizar la tarea a partir de su experiencia como diseñador educacional, basándose en su conocimiento del contenido del *CdC*.

Resultados del usuario Usuario 2

Alejandro	Tiempo (s)	Logrado (sí/no)	CA asociado
<i>Tarea 1</i>	145	Sí	DE-2, DE-7, DE-9, DE-10, DE-11
<i>Tarea 2</i>	20	Sí	DE-1, DE-5, DE-6
<i>Tarea 3</i>	125	Sí, con dificultad	DE-3, DE-8, DE-15
<i>Tarea 4</i>	- (*)	Sí	DE-14, DE-16
<i>Tarea 5</i>	50	Sí	DE-12, DE-19, DE-21, DE-22, DE-23

Tabla 5.7: Tabla de resultados del estudio de usabilidad del *Issue Tracker* y el *MoME* para el usuario Diseñador Educacional, realizado por el usuario 2. (*) Ver tarea 4.

Comentarios del usuario 2:

- Con respecto a la tarea 1, el usuario declara que la tarea no está siendo realizada dado que no está asignada a algún miembro del equipo.
- Con respecto a la tarea 3, el usuario reconoce los recursos a partir de las columnas del *MoME*. No obstante, dado que no existe un acceso directo a los contenidos que deben ser modificados, este los encuentra utilizando su experiencia como diseñador educacional.

- Con respecto a la tarea 5, el usuario advierte que al marcar la tarea como completada, se está saltando un paso de verificación.

5.3.3. Análisis de los resultados

En la tablas 5.8 y 5.9 se resumen los resultados de los estudios realizados por ambos usuarios, para luego entrar en el análisis de estos. En primer lugar se busca interpretar los resultados obtenidos, para luego entrar en detalles basados en los comentarios de los usuarios entrevistados.

Métrica	Usuario 1	Usuario 2
Tareas por realizar	5	
Tareas completadas con éxito	5	5
Éxito de tarea 1	100 %	
Tiempo tarea 1	210	145
Tiempo promedio tarea 1 (segundos)	177.5	
Éxito de tarea 2	100 %	
Tiempo tarea 2 (segundos)	15	20
Tiempo promedio tarea 2 (segundos)	17.5	
Éxito de tarea 3	100 %	
Tiempo tarea 3 (segundos)	220	125
Tiempo promedio tarea 3 (segundos)	172.5	
Éxito de tarea 4	100 %	
Tiempo tarea 4 (segundos)	-	-
Tiempo promedio tarea 4 (segundos)	-	
Éxito de tarea 5	100 %	
Tiempo tarea 5 (segundos)	70	50
Tiempo promedio tarea 5 (segundos)	60	
Tiempo total del estudio (segundos)	720	480

Tabla 5.8: Tabla recapitulativa de las tareas y los tiempos obtenidos en el estudio de usabilidad del uso conjunto de las herramientas *MoME* e *Issue Tracker* por dos usuarios diseñadores educacionales. Nótese que el tiempo en la tarea 4 no fue tomado en cuenta, esto se justifica en la descripción de la tarea 4.

Lo primero a notar en la tabla 5.8, es que el 100% de las tareas fueron realizadas. No obstante, las tareas 1 y 3 fueron realizadas con dificultad. Esto está asociado a principalmente al nivel de práctica de los usuarios que realizaron el estudio: el usuario 1 utilizaba el conjunto de herramientas por primera vez, y fue necesario guiarlo con pistas para identificar elementos presentes dentro de la interfaz; el usuario 2 tenía algo de experiencia utilizando la herramienta, pero la última vez que la había usado fue aproximadamente 3 meses atrás, además de considerar que llevaba cerca de 40 días fuera por vacaciones y formaciones realizadas en el extranjero. En otras palabras, ambos usuarios carecían de práctica y/o experiencia al utilizar el conjunto de herramientas.

Por otro lado, la disparidad de práctica justifica también la diferencia de tiempos entre ambos usuarios, dónde el usuario 2 realiza cada tarea con entre un 25 % a 44 % menos tiempo que el usuario 1. En términos del tiempo total, esta diferencia representa entre 0.7-16 % del tiempo total, lo que es una diferencia significativa. En otras palabras, esto indica que a partir de la experiencia es posible reducir los tiempos necesarios para llevar a cabo el flujo de cambio utilizando el conjunto de herramientas *MoME* e *Issue Tracker*.

Además, la tarea 3, correspondiente a ubicar dentro de los soportes las páginas y diapositivas específicas por modificar, no posee actualmente un acceso directo a partir de ninguna de las dos herramientas. La razón principal para ello es el carácter incierto del estado de los soportes, dada la gran cantidad de modificaciones que se les realizan. A modo de ejemplo, si se indicase una página específica para realizar una modificación, y resulta que a raíz de otra modificación, esa página cambia, entonces la información estaría incorrecta, lo que induciría a un error o a confusión al usuario. Es por ello que por ahora se prefiere nombrar los elementos dentro del *MoME*, más que indicar un lugar preciso. No obstante, el poder referenciar los elementos de manera única es parte de los desafíos a resolver para el futuro de la solución.

Con respecto a la tarea 5, el tiempo requerido está asociado a que los usuarios no recordaban/sabían dónde se encontraba el componente *Trello for MoME* de *Google Spreadsheet* desarrollado en este trabajo, por lo que fue necesario indicarles dónde se encontraba. Cabe resaltar que el usuario 2, destaca que la herramienta se salta el paso de validación, al enviar las tarjetas directamente al estado *Done* del *Issue Tracker*. No obstante, la columna *Check*, correspondiente a la validación existe, y se considerará su uso para los pasos siguientes de este trabajo.

Métrica	Usuario 1	Usuario 2
Número total de CAs para los facilitadores	27	
Número de CAs evaluados	18	
Número de CAs validados	16	16
Porcentaje de CAs evaluados c/r al total	67 %	
Porcentaje de CAs validados c/r al total	59 %	59 %
Porcentaje CAs validados c/r a los evaluados	89 %	89 %

Tabla 5.9: Tabla recapitulativa de los porcentajes de CAs evaluados y validados en el estudio de usabilidad del uso conjunto de las herramienta del *MoME* e *Issue Tracker* por dos usuarios diseñadores educacionales.

Con respecto a la tabla 5.9, es posible afirmar que de los 27 CAs relacionados con el diseñador educacional, 18 fueron evaluados, lo que represente 67 % del total, es decir dos tercios. De la misma manera, 16 de los CAs fueron validados por el estudio, es decir 59 % del total. Considerando solamente los CAs evaluados, es posible afirmar que el 89 % de los criterios fue validado durante el estudio. Cabe resaltar que ese último porcentaje corresponde a los CAs asociados a tareas críticas del sistema, por lo que es bastante buen resultado en cuanto al comportamiento esperado.

5.3.4. Diseño de la evaluación utilizando la System Usability Scale (SUS)

El diseño de la evaluación es idéntica a aquél presentado en la subsección 5.2.4.

5.3.5. Ejecución de la evaluación

Para llevar a cabo la evaluación de usabilidad, se utiliza un *Google Forms* con todas las aseveraciones y la posibilidad de responder dada la escala del 1 al 5 presentada en la sección anterior. Este formulario estaba vinculado a un *Google Spreadsheet*, desde el cuál se calcula el índice de usabilidad. En las tablas 5.5 se encuentran los resultados de los dos usuarios entrevistados.

5.3.6. Análisis de los resultados

Aseveración	Puntaje usuario 1	Puntaje usuario 2
1. <i>Creo que usaría este sistema frecuentemente</i>	3	4
2. <i>Encuentro este sistema innecesariamente complejo</i>	3	4
3. <i>Creo que el sistema fue fácil de usar</i>	3	3
4. <i>Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar este sistema</i>	2	4
5. <i>Las funciones de este sistema están bien integradas</i>	3	2
6. <i>Creo que el sistema es muy inconsistente</i>	3	4
7. <i>Imagino que la mayoría de la gente aprendería a usar este sistema en forma rápida</i>	2	4
8. <i>Encuentro que el sistema es muy difícil de usar</i>	2	2
9. <i>Me siento confiado al usar el sistema</i>	4	2
10. <i>Necesité aprender muchas cosas antes de ser capaz de usar este sistema</i>	3	4
Puntuación SUS (sobre 100)	55	42.5
Puntuación promedio SUS	48.75	

Tabla 5.10: Tabla con los resultados de la encuesta cualitativa basada en SUS realizado a dos usuarios.

Los resultados de la evaluación se presentan en la tabla 5.10. De ellos es posible observar que la mayoría de las aseveraciones tiene puntajes similares, es decir con a lo más 1 punto

de distancia, a excepción de las aseveraciones 4, 7 y 9. Esto indica que la percepción de los usuarios es parecida a excepción de dichos puntos.

Al indagar en los resultados de las aseveración 4, se observa que el usuario 1 no cree necesitar ayuda de una persona con conocimientos técnicos, indicando un puntaje de 2, mientras que el usuario 2 responde con un 4 a la misma aseveración. Esto no nos permite concluir sobre dicho punto, dado que al promediar ambos resultados se obtiene un promedio neutral de 3.

Con respecto a la aseveración 7, ocurre algo similar a la aseveración 4, el usuario 1 no cree que la mayoría de la gente aprendería a utilizar el sistema de forma rápida, con una puntuación de 2, mientras que el usuario 2 responde con un 4 a la misma aseveración. En otras palabras no es posible concluir al respecto.

La aseveración 9 posee la misma diferencia de 2 puntos, dónde el usuario 1 se siente confiado al usar el sistema, con una puntuación de 4, mientras que el usuario 2 no tanto, con una puntuación de 2.

Estas tres aseveraciones denotan algo curioso: el usuario 1, quien se enfrentaba a la herramienta por primera vez, dice sentirse más confiado, necesitar de menos ayuda y creer que el sistema será de fácil adopción por otros, mientras que el usuario más experimentado expresa lo contrario. Del usuario 1, es posible deducir que al sentirse confiado, y ser su primer uso, el sistema se asemeja al proceso previo a este trabajo, por lo que se cumple con la característica de calidad de adaptabilidad de la solución.

En lo que respecta el resto de las aseveraciones, la mayoría se encuentran en promedio cercanas al 3, lo que es bastante neutral, por lo que no es posible concluir mucho al respecto.

Por otro lado, la puntuación total en una escala de 100 es de 55 y 42.5 puntos respectivamente para el usuario 1 y 2, siendo un promedio de 48.75, lo cual está 19.25 puntos más abajo del puntaje asociado a un sistema con buena usabilidad, es decir 68 puntos, según los autores de la herramienta. En conclusión ambos usuarios estiman que el sistema tiene mala usabilidad, por lo que es necesario mejorarla. Esto podría explicarse dado el hecho de que el conjunto de herramientas no está totalmente integrado al flujo de trabajo de los usuarios, y requiere de trabajo extra para ser utilizado. En otras palabras el foco se encuentra en integrar ambas herramientas con el navegador para llegar a una experiencia usuario y usabilidad mejores.

5.4. Resumen

Finalizado este capítulo, el lector posee una visión global de las validaciones intermedias, semejantes a pruebas unitarias, además de las validaciones finales, o pruebas de integración. En particular, se valida el uso de versionamiento para distinguir elementos del *CdC*; la pertinencia del uso del *IT* y del contenido del *MoME*, contrastado a la dificultad de uso, con una puntuación de usabilidad de 48.75 puntos; la utilidad del *Navegador* y su buena usabilidad, ligada a un puntaje de 72.5 puntos.

Capítulo 6

Conclusiones

En este capítulo se busca concluir con respecto al trabajo realizado. Para ello se realiza un breve resumen de los objetivos, metodologías, desarrollo y validación de lo realizado; se plantea aquello que no funcionó, partiendo desde lo más general a lo más particular; se dan recomendaciones en caso de que alguien quiera seguir trabajando en este proyecto, en particular si vale la pena o no seguir trabajando en él; recomendaciones para memorias similares para finalmente exponer los aprendizajes que el autor se lleva de este trabajo.

6.1. Resumen del trabajo

El problema a resolver consistía en la gestión de los cambios del contenido del *CdC* de la empresa *LeanSight*, teniendo como objetivo el de identificar, estructurar y proveer las herramientas de software para apoyar dicho proceso.

Lo anterior se realizó utilizando el ciclo de Deming, que consiste en ciclos cortos, compuestos por cuatro etapas: planificación, es decir plantear los objetivos del ciclo, y cómo se va a realizar; acción, o llevar a cabo lo propuesto; verificación de si se cumplen los objetivos o no; ajuste de los objetivos a partir de los resultados de la verificación, para finalmente volver a comenzar el ciclo con mayor información que previamente. Desde el punto de vista de las metodologías, se utiliza *Genba*, para vivir los procesos y entender a priori su funcionamiento, acto seguido se busca llegar al problema de causa raíz, a través de los *5 porqués*, buscando la relación causal de las observaciones y síntomas observados durante la vivencia del proceso. Esto se inscribe dentro del *Análisis de Causa Raíz*, el cual tiene por objetivo proponer soluciones a partir de la información recopilada. Las soluciones propuesta son luego revisadas por un maestro o experto que conoce más del tema y que valida si es necesario recopilar más información o ya es posible implementar la solución. A esto se le conoce como *Nemawashi*, metodología que habilita el diseño iterativo de la solución, característica deseada de calidad. Nótese que si bien existen validaciones intermedias en cada ciclo de Deming, se realizó una validación final a través de un estudio de usabilidad de las herramientas desarrolladas, además de utilizar la Escala de Usabilidad de un Sistema (*EUS*) o *System Usability Scale (SUS)* para obtener un índice de usabilidad de estas últimas, lo cual está enfocado en medir cuán

buena es la experiencia usuario, criterio de calidad deseado.

Teniendo en cuenta la metodología utilizada, el objetivo fue separado en cinco objetivos específicos, el primero consistía en modelar la situación actual, para poder identificar los problemas presentes, lo cual fue realizado en sección 3.1, y que permitió establecer un análisis de causa raíz del problema, y por ende entender el problema de fondo además de los procesos involucrados.

El segundo objetivo específico apuntaba a diseñar, implementar y validar un flujo estructurado de gestión del cambio para módulos de aprendizaje compuestos de múltiples documentos. Para ello, se implementaron versiones que permitiesen distinguir entre distintas instancias de módulos y sesiones. Acto seguido, se buscó poder caracterizar los problemas encontrados: adaptando la información necesaria para caracterizar incidentes en los sistemas informáticos llamados *Issue Trackers*, fue posible identificar la información necesaria para poder denotar un problema. Si bien era posible caracterizar un problema, faltaba poder identificar qué elemento debía ser modificado. Para ello, se modeló el relato lógico de un módulo de aprendizaje, resultando en el *MoME*, a partir del cual es posible identificar de manera específica qué elemento(s) es necesario modificar. Finalmente, fue necesario proponer un flujo de cambio adaptado a lo anterior. Basándose en el sistema de versionamiento *Git*, el funcionamiento de un *Issue Tracker* y el flujo de *Discovery - Delivery* se propuso e implementó un flujo estructurado de cambio a través del uso de un *Issue Tracker* adaptado a la situación de la empresa y el proceso de cambio. En otras palabras, es posible realizar modificaciones al contenido del *CdC*, a partir de observaciones y pasando por caracterizar el problema en forma de incidente, asociarlo a un elemento del relato, para luego realizar el seguimiento de dicho incidente, hasta realizada la modificación, dejando listo el contenido para su uso.

El tercer objetivo específico consiste en diseñar, implementar y validar un flujo estructurado que permita acceder a las distintas versiones de los módulos y/o sesiones realizadas. Para ello, se implementó un navegador de sesiones y módulos, que permite visualizar de manera ordenada las sesiones y módulos, pudiendo acceder a estos contenidos, crear sesiones y módulos además de publicarlos. En otras palabras, el navegador viene a completar la parte final del proceso de cambio, al habilitar la creación y publicación de las sesiones en un formato pertinente para realizar formaciones. A partir de estas, los facilitadores son capaces de observar cuan bien se logran transmitir los contenidos a los aprendices, además de obtener retroalimentación de estos últimos. Ambas instancias permiten identificar problemas que serán caracterizados como incidentes, para volver a empezar el proceso de cambio.

El cuarto objetivo específico buscaba apoyar el proceso de gestión de cambios con herramientas de software para minimizar la probabilidad de inconsistencia entre documentos, los cambios sin razón aparente y la pérdida de datos. Como vimos en los párrafos anteriores, las herramientas *MoME* e *Issue Tracker* permiten minimizar la probabilidad de pérdidas de datos, además de dar razón a los cambios, al visibilizar el lugar, razón y estado de las modificaciones por realizar. En lo que respecta al *Navegador*, este permite acceder a todas las sesiones y módulos realizados con anterioridad, por lo que fueron puestos a prueba y por ende se minimiza la probabilidad de que estos sean inconsistentes. En otras palabras se logra cumplir con este objetivo específico.

El quinto y último objetivo específico, consiste en proveer una buena experiencia usuario

(UX) en el uso conjunto de herramientas, de tal manera de asegurarse que estas sean utilizadas. Para verificar si se cumplió, fue necesario llevar a cabo una validación del producto final, por lo que se realizó un estudio de usabilidad y se midió el índice de usabilidad utilizando *SUS* de dos cosas: el uso conjunto del *MoME* e *Issue Tracker*, y el uso del *Navegador*. Ambos estudios de usabilidad revelaron que los usuarios podían usar los sistemas, al poder realizar la gran mayoría de las tareas indicadas. En lo que respecta la escala de usabilidad del sistema, el *Navegador* obtuvo un índice de usabilidad de 70 en promedio, considerado bueno; el uso conjunto del *MoME* e *Issue Tracker* obtuvo un índice de usabilidad de 48,75 en promedio, considerado malo.

En lo que respecta a la empresa, este trabajo habilitó el explicitar un gran número de prácticas implícitas, lo cual permitió visibilizar y luego gestionar el trabajo de diseño y rediseño del contenido del *CdC*. Además, si bien las herramientas tecnológicas implementadas en este trabajo no son del todo usables, si poseen potencial, según la opinión de los colaboradores. En otras palabras, es un trabajo que se debe seguir.

En conclusión, se logra cumplir con todos los objetivos específicos de este trabajo, lo que en conjunto permite cumplir con el objetivo general. No obstante, las herramientas de software implementadas tienen índices de usabilidad variados: en el caso del *Navegador*, se puede hablar de una herramienta usable, por lo que los facilitadores se verán beneficiados por su uso; por otro lado, el uso conjunto del *MoME* e *Issue Tracker* no se considera un sistema usable, por lo que sería necesario mejorar la experiencia usuario de dichas herramientas. En lo que respecta el aporte a la empresa *LeanSight*, el trabajo tiene potencial, pero le falta usabilidad, por lo que se debe seguir trabajando en el proyecto.

Para dar cuenta de lo que no se logró, es posible basarse en principalmente dos cosas: los criterios de aceptación no validados, y los resultados de la validación final, en particular el índice de usabilidad de la *SUS*.

En lo que respecta los Criterios de Aceptación, se plantea un total de 40, asociados a los distintos roles de los colaboradores. De los 27 CAs asociados a los Diseñadores educacionales, se validan un total de 16 usando un estudio de usabilidad, lo que representa un 59 % del total. En el caso de los facilitadores, de los 12 CAs asociados, se valida un total de 7 con un estudio de usabilidad, lo que representa un 64 % del total. En otras palabras, se valida un total de 23 CAs de los 40 que hay, o un 57,5 %, correspondientes a las tareas críticas del sistema. Cabe resaltar, que 12 CAs no fueron evaluados, lo que implica que de los 28 CAs evaluados, 23 son validados, lo que corresponde a un 88,5 % de validación de las tareas críticas.

Lo anterior implica que 5 CAs fueron rechazados, estos corresponden, en el caso del diseñador educacional, principalmente a la inhabilidad de agrupar cambios de manera coherente, además de no poder navegar hasta el elemento específico de un soporte que debe ser modificado sin conocimiento del *CdC*. En el caso del facilitador, este no es capaz de diferenciar distintas instancias de los módulos a través del uso del navegador.

Si se consideran ahora los CAs que no fueron evaluados a través de un estudio de usabilidad, las validaciones intermedias permiten entregar información sobre si estos fueron validados o no. En el caso de los diseñadores educacionales, una parte corresponde al uso del *Issue Tracker*, es decir: poseer la capacidad de priorizar entre distintas modificaciones y poseer la

información suficiente sobre el cambio a realizar, lo que corresponde a poder leer un flujo de *Kanban*, práctica instaurada dentro de la empresa, por ende validada. La otra parte corresponde a poseer los elementos necesarios para construir módulos, lo cual es en parte realizado al haber automatizado la creación de módulos a través del navegador. Sin embargo, falta implementar las operaciones de actualización y borrado para poder dar total libertad al usuario al respecto. En el caso de los facilitadores, los CAs no evaluados corresponden a poder entender la lógica del relato del módulo, lo que se hace a través del *MoME*; saber qué información es necesaria para caracterizar un cambio, lo que corresponde a la utilización del *Issue Tracker*, y finalmente a la posibilidad de poder compartir los contenidos de una sesión con el público asociado, lo cual si bien no está automatizado, es posible. En lo que respecta a los CAs asociados tanto a los diseñadores educacionales como los facilitadores, estos son dos, y corresponden a conocer los pasos a seguir para realizar el proceso de cambio, lo que está validado a través del uso de las herramientas, además de conocer la estructura y orden que deben seguir los contenidos de los soportes dentro de la unidad de almacenamiento, criterio que deja de ser necesario al abstraer el manejo y orden de los documentos utilizando el navegador.

Tomando en cuenta todo lo anterior, de los 40 criterios de aceptación, hay 23 validados y 5 rechazados por estudios de usabilidad, 8 validados y 4 rechazados por validaciones intermedias. En otras palabras, hay 31 CAs validados, lo que representa un 77,5 % del total.

En lo que respecta los criterios de calidad deseados, el hecho de tener un diseño incremental que permitiese tanto utilizar como validar las soluciones implementadas se logra por medio de la aplicación del ciclo de Deming, inscrito dentro de la metodología *Nemawashi*. Por otro lado, el tema de la experiencia usuario, no se logró totalmente, al tener índices de usabilidad variados: el navegador se considera usable, mientras que el uso conjunto de las herramientas *MoME* e *Issue Tracker* no son considerados usables. En otras palabras es necesario seguir trabajando la usabilidad y la experiencia usuario.

6.2. Trabajo futuro

Tal y como se plantean en los apartados anteriores, si bien el trabajo cumplió con los objetivos planteados y con gran parte de los criterios de aceptación, la usabilidad deja que desear en el caso del uso conjunto del *MoME* e *Issue Tracker*, por lo que gran parte del trabajo futuro debe considerar una mejora en la usabilidad de dichas herramientas. Además, se considera importante trabajar en la integración de todas las herramientas para apoyar a los usuarios durante la totalidad del proceso de cambio. Para ello se recomienda seguir utilizando iteraciones cortas y probando soluciones e implementaciones hasta encontrar aquellas que se adapten a los usuarios.

A modo de resumen, es necesario enfocarse en la integración de todas las herramientas, trabajando en conjunto con los usuarios para poder entregar la mejor experiencia usuario.

6.3. Aprendizajes sobre el trabajo de título

Los aprendizajes de este trabajo son muchos, por lo que se intentará resumirlos.

En caso de escribir una memoria similar, es decir un trabajo con ciclos cortos y enfocado en resolver un problema de negocios de una empresa, se recomienda considerar siempre la opinión del usuario al proponer soluciones, y esperar a tener su validación antes de implementarlas. Esto dado que cambiar una solución es mucho menos costoso que cambiar una implementación, sin considerar el tiempo utilizado en implementar y posterior retrabajo que eso significa.

Con respecto a los ciclos cortos, es posible afirmar que las validaciones intermedias permiten ajustar de mejor manera la solución al usuario, y que sin ellas hubiese sido difícil llegar a la solución actual. Se recomienda fuertemente aplicar el ciclo de Deming con ciclos cortos para poder ajustar la solución en el camino y verificar luego de cada ciclo cuál es el foco del trabajo, el cual puede cambiar drásticamente en función de la nueva información que genera cada ciclo.

Por otro lado, dado que la solución se va “construyendo en el camino”, confiar en que se está haciendo lo mejor que se puede con la información y habilidades que se tiene, y no tener miedo a modificar lo implementado, dado que si bien una solución puede haber significado mucho trabajo y cuesta “desecharla”, no es realmente una pérdida, si no que un aprendizaje.

Otro de los aprendizajes es que un trabajo de memoria requiere de trabajo constante, escribir los resultados que se obtienen es parte fundamental de ello, dado que entrega claridad sobre lo que se está haciendo. Si bien existe la posibilidad de que cierta nueva información cambie en gran parte la manera de entender el trabajo, la información ya escrita permite visibilizar lo que se tiene y poder adaptarlo más fácilmente. En otras palabras la escritura del documento es un proceso incremental, con muchas instancias de refactorización de los contenidos.

Otra cosa importante es que la experiencia de trabajar con una empresa es sumamente valiosa. De partida permite entrar al mundo laboral con un poco de anterioridad, y por ende crecer profesionalmente y construir curriculum, lo que ayuda a la empleabilidad a futuro. Sin embargo, lo más importante está en poder sumergirse en un ambiente controlado de trabajo, al poseer la estructura de la universidad por un lado y los intereses de la empresa por otro. Entendiendo que uno realiza un trabajo que se intersecta en muchas oportunidades, también es necesario considerar el trabajo extra que cada parte requiere que se realice. Esto puede ser difícil de manejar, pero la comunicación y transparencia con ambas partes es fundamental para poder avanzar y cumplir con las expectativas de todos. Para ello es necesario elegir bien con quién asociarse, y quizás priorizar un ambiente de crecimiento personal, con buenos mentores que den las herramientas para ello.

Si bien el documento de memoria es un trabajo tedioso, es parte del proceso de titulación, y es una oportunidad de realizar un proyecto de gran envergadura por primera vez, en un ambiente controlado, donde es posible fallar en pequeño. La posibilidad de fallar es un concepto que se aleja bastante de lo que se enseña en la escuela, y que creo es necesario para la

vida: no siempre se va a ser exitoso en todo, y uno va a caerse en muchas oportunidades, lo importante es volver a levantarse, y ver de qué manera es posible hacer mejor las cosas, para seguir creciendo, tanto como persona y como profesional.

Podría extenderme mucho más, pero creo que estos tres puntos cubren suficiente los aprendizajes que obtuve de este trabajo. Finalizo este documento diciendo que si bien uno puede verse llamado por incentivos financieros, aprendí que es mejor priorizar ambientes de aprendizaje y desarrollo personal/profesional, junto a un equipo donde sea un agrado trabajar, por lo que recomiendo fuertemente seguir el propio instinto y elegir un lugar agradable de trabajar, donde exista a seguridad emocional para fallar, entendiendo que siempre se está dando lo mejor de sí.

Bibliografía

- [1] Nerea Boada. ¿por qué es tan importante el user experience o experiencia del usuario? <https://www.cyberclick.es/numerical-blog/por-que-user-experience-o-experiencia-del-usuario>. Último acceso: 12 de Julio 2019.
- [2] John Brooke. “*SUS-A quick and dirty usability scale*” *Usability evaluation in industry*. CRC Press, June 1996. ISBN: 9780748404605.
- [3] Ben Collins-Sussman, Brian W. Fitzpatrick, and C. Michael Pilato. *Version Control with Subversion*. O’Reilly, 2002.
- [4] Datanyze. Github market share. <https://www.datanyze.com/market-share/source-code-management/github-market-share>. Último acceso: 12 de Julio 2019.
- [5] Vincent Driessen. A successful git branching model. <https://nvie.com/posts/a-successful-git-branching-model/>. Último acceso: 12 de Julio del 2019.
- [6] David Fuller, Sergio Mujica, and José A. Pino. The design of an object-oriented collaborative spreadsheet with version control and history management. pages 416–423, 01 1993.
- [7] Google. *CLASP GitHub Repository*. Último acceso: 12 de Julio 2019.
- [8] Google. Google apps scripts. <https://developers.google.com/apps-script/>. Último acceso: 12 de Julio 2019.
- [9] Google. Google drive. https://www.google.com/intl/es_ALL/drive/. Último acceso: 12 de Julio 2019.
- [10] J.M. Gross and K.R. McInnis. *Kanban Made Simple: Demystifying and Applying Toyota’s Legendary Manufacturing Process*. Number v. 2 in *Kanban Made Simple: Demystifying and Applying Toyota’s Legendary Manufacturing Process*. AMACOM, 2003.
- [11] M. Imai. *Gemba Kaizen: A Commonsense, Low-Cost Approach to Management*. Mcgraw-hill, 1997.
- [12] The W. Edwards Deming Institute. Pdsa cycle. <https://deming.org/explore/p-ds-a>. Último acceso: 12 de Julio 2019.

- [13] LeanSight. Leansight encontrando soluciones alternativas a través del desafío focal. <https://www.youtube.com/watch?v=x-irzSZezsA>. Último acceso: 12 de Julio 2019.
- [14] Microsoft. Visual studio code. <https://code.visualstudio.com/>. Último acceso: 12 de Julio 2019.
- [15] Nested Public Modules. Tslint. <https://www.npmjs.com/package/tslint>. Último acceso: 12 de Julio 2019.
- [16] Jakob Nielsen. Thinking aloud: The #1 usability tool. <https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/>. Último acceso: 12 de Julio 2019.
- [17] Art of Lean. Is it genba or gemba? <http://artoflean.com/index.php/2016/03/25/is-it-genba-or-gemba/>. Último acceso: 12 de Julio 2019.
- [18] BenStraub Scott Chacon. Git. <https://git-scm.com/about>. Último acceso: 12 de Julio 2019.
- [19] Olivier Serrat. *The Five Whys Technique*. 05 2017.
- [20] John Shook. Managing to learn: Using the a3 management process to solve problems, gain agreement, mentor and lead. https://www.amazon.com/dp/1934109207/ref=cm_sw_r_other_ap_a_i_sN-HCbB7V1XD8. Último acceso: 12 de Julio 2019.
- [21] Mind Tools. 5 whys. https://www.mindtools.com/pages/article/newTMC_5W.htm. Último acceso: 12 de Julio 2019.
- [22] Trello. Trello official website. <https://trello.com>. Último acceso: 12 de Julio 2019.
- [23] Elijah Valenciano. The acceptance criteria for writing acceptance criteria. <https://medium.freecodecamp.org/the-acceptance-criteria-for-writing-acceptance-criteria-6eae9d497814>. Último acceso: 12 de Julio 2019.
- [24] Agustín Villena. Descubriendo el impacto de negocio detrás de los proyectos tecnológicos. <https://medium.com/mejorindustriati/descubriendo-el-impacto-de-negocio-detr%C3%A1s-de-los-proyectos-tecnol%C3%B3gicos-85e50984d622>. Último acceso: 20 de agosto del 2019.
- [25] Agustín Villena. Un modelo empírico de enseñanza de las metodologías ágiles: El caso del curso cc62v –taller de metodologías ágiles de desarrollo de software. Master’s thesis, Universidad de Chile, Departamento de Ciencias de la Computación, 2008.
- [26] W3School. Javascript gettime() method. https://www.w3schools.com/jsref/jsref_gettime.asp. Último acceso: 12 de Julio 2019.
- [27] P.F. Wilson, L.D. Dell, and G.F. Anderson. *Root Cause Analysis: A Tool for Total Quality Management*. ASQC Quality Press, 1993.
- [28] xplaner. Google apps what is visual thinking? <http://www.xplaner.com/visual->

thinking-school/. Último acceso: 12 de Julio 2019.

- [29] Jeff Zyches. Discovery kanban at optimizely. <https://medium.com/design-optimizely/discovery-kanban-at-optimizely-7b3025066d54>. Último acceso: 12 de Julio 2019.

Apéndice A

Criterios de Aceptación

A continuación se encuentran todos los criterios de aceptación de la solución estos se encuentran codificados en función del autor del criterio. En otras palabras si un Diseñador Educativo (DE) debe poder realizar una cierta acción, entonces el criterio de aceptación comienza por las siglas DE-X, dónde X denota el número del criterio. En particular:

- DE-X: Diseñador Educativo
- DEF-X: Diseñador Educativo o Facilitador
- F-X: Facilitador

DE-1 Yo, como diseñador educativo de un módulo,
cuando termino de diseñar el módulo,
entonces quiero poder comunicar la visualización del relato con el resto de los colaboradores a través de algún soporte.

DE-2 Yo, como, diseñador educativo buscando mejorar el contenido del *CdC*
cuando visualizo el relato a través de un soporte,
entonces soy capaz de identificar en qué lugares existe necesidad de cambio.

DE-3 Yo, como, diseñador educativo buscando mejorar el contenido del *CdC*
cuando visualizo el relato a través de un soporte,
entonces soy capaz de identificar qué relaciones existen entre las distintas actividades pedagógicas.

DE-4 Yo, como diseñador educativo buscando mejorar el contenido del *CdC*,
cuando visualizo el relato a través de un soporte,
entonces soy capaz de priorizar entre las distintas modificaciones por realizar.

DE-5 Yo, como diseñador educativo habiendo identificado un problema,
cuando lo caracterizo con toda la información pertinente,
entonces tengo un soporte dónde poner dicha información que el resto de los diseñadores educativos pueden acceder.

DE-6 Yo, como diseñador educativo queriendo realizar cambios,

cuando consulto el soporte,
entonces existe una razón concreta que justifica el cambio.

DE-7 Yo, como diseñador educacional queriendo realizar cambios,
cuando consulto el soporte,
entonces sé qué cambios se deben realizar.

DE-8 Yo, como diseñador educacional queriendo realizar cambios,
cuando consulto el soporte,
entonces sé dónde se debe realizar cada cambio.

DE-9 Yo, como diseñador educacional queriendo realizar cambios,
cuando consulto el soporte,
entonces sé si algún otro diseñador educacional está realizando un cambio en particular y que por ende no debo modificar el mismo contenido.

DE-10 Yo, como diseñador educacional queriendo realizar cambios,
cuando consulto el soporte,
entonces sé qué cambios ya tuvieron lugar.

DE-11 Yo, como diseñador educacional queriendo realizar cambios,
cuando consulto el soporte,
entonces sé en qué estado se encuentra cada cambio.

DE-12 Yo, como diseñador educacional realizando cambios,
cuando finalizo un cambio,
entonces puedo denotar toda la información pertinente a dicho cambio en el soporte.

DE-13 Yo, como diseñador educacional queriendo verificar la correctitud de los cambios,
cuando consulto el soporte,
entonces poseo la información pertinente para aprobar o rechazar dicho cambio.

DE-14 Yo, como diseñador educacional queriendo realizar cambios,
cuando decido realizar un cambio específico,
entonces soy capaz de visualizar todos los soportes asociados a dicho cambio.

DE-15 Yo, como diseñador educacional queriendo realizar cambios,
cuando selecciono un cambio específico,
entonces soy capaz de navegar en menos de 5 clicks a la ubicación aproximada del cambio dentro de los soportes asociados.

DE-16 Yo, como diseñador educacional queriendo realizar cambios,
cuando realizo un cambio específico,
entonces soy capaz de visualizar si existe propagación del cambio en otro soporte.

DE-17 Yo, como diseñador educacional,
cuando realizo cambios,
entonces sé identificar lo que es un conjunto coherente de cambios.

- DE-18 Yo, como** diseñador educacional,
cuando realizo un conjunto coherente de cambios,
entonces soy capaz de agruparlos a través de algún soporte.
- DE-19 Yo, como** diseñador educacional,
cuando agrupo un conjunto coherente de cambios,
entonces soy capaz de asociarlos a una razón específica que viene a describir dicho conjunto.
- DE-20 Yo, como** diseñador educacional,
cuando agrupo y describo un conjunto coherente de cambios,
entonces soy capaz de declarar un punto estable.
- DE-21 Yo, como** diseñador educacional habiendo identificado un problema,
cuando quiero caracterizarlo,
entonces soy capaz de hacerlo desde un centro de control.
- DE-22 Yo, como** diseñador educacional,
cuando inicio el proceso de cambio,
entonces quiero poder realizar todas las etapas desde un mismo soporte.
- DE-23 Yo, como** diseñador educacional habiendo realizado una modificación,
cuando quiero denotar que esta fue realizada,
entonces soy capaz de hacerlo desde un centro de control.
- DE-24 Yo, como** diseñador educacional, independiente de mi formación profesional,
cuando sigo el proceso de cambio,
entonces soy capaz de llevar a cabo las tareas asociadas al proceso de cambio.
- DE-25 Yo, como** diseñador educacional,
cuando construyo un módulo,
entonces sé cuales son los elementos necesarios para hacerlo.
- DE-26 Yo, como** diseñador educacional,
cuando construyo un módulo,
entonces tengo a mi disposición los modelos (templates) de los elementos necesarios.
- DE-27 Yo, como** diseñador educacional,
cuando construyo un módulo,
entonces soy capaz de crear los modelos (templates) del elemento que necesito de manera automática.
- DEF-1 Yo, como** diseñador educacional o facilitador,
cuando identifico un problema,
entonces conozco los pasos a seguir para realizar el proceso de cambio.
- DEF-2 Yo, como** diseñador educacional o facilitador,
cuando modifico el orden de los soportes dentro de la unidad de almacenamiento de la compañía,

entonces sé cual es la estructura y orden que debiesen seguir.

- F-1 Yo, como** facilitador que no participó del proceso de diseño de un módulo en específico,
cuando los diseñadores educacionales responsables de dicho módulo terminan de diseñarlo,
entonces quiero ser capaz de entender la lógica del relato del módulo en específico a partir de un soporte.
- F-2 Yo, como** facilitador realizando un taller,
cuando identifico un posible problema o necesidad de cambio,
entonces quiero saber qué información es pertinente escribir para caracterizar dicho problema.
- F-3 Yo, como** facilitador,
cuando poseo los soportes en un estado estable,
entonces soy capaz de identificar en qué momento debo publicar.
- F-4 Yo, como** facilitador,
cuando voy a realizar un taller en el futuro cercano,
entonces soy capaz de publicar los soportes en el formato apropiado para realizar formaciones.
- F-5 Yo, como** facilitador,
cuando quiero publicar los soportes en el formato apropiado para realizar formaciones,
entonces lo puedo hacer de manera automática desde un centro de control.
- F-6 Yo, como** facilitador,
cuando publico los contenidos,
entonces soy capaz de distinguir las distintas instancias de contenidos publicados.
- F-7 Yo, como** facilitador,
cuando accedo a una sesión de un taller,
entonces soy capaz de distinguirlo de otra sesión del mismo taller.
- F-8 Yo, como** facilitador,
cuando distingo entre sesiones,
entonces soy capaz de hacerlo a partir del contenido de este.
- F-9 Yo, como** facilitador,
cuando distingo entre sesiones,
entonces soy capaz de hacerlo a partir del público para el cuál fue realizado.
- F-10 Yo, como** facilitador habiendo realizado una formación a un público específico,
cuando es momento de compartir los soportes asociados a dicha sesión,
entonces soy capaz de compartir los soportes asociados a la sesión al público correspondiente.

- F-11 Yo, como** facilitador,
cuando distingo entre sesiones,
entonces soy capaz de comparar entre ellas en función del público, contenido.
- F-12 Yo, como** facilitador,
cuando identifico los contenidos necesarios para una sesión desde el navegador,
entonces soy capaz de crear una nueva sesión.

Apéndice B

Diagrama general detallado de la
situación actual

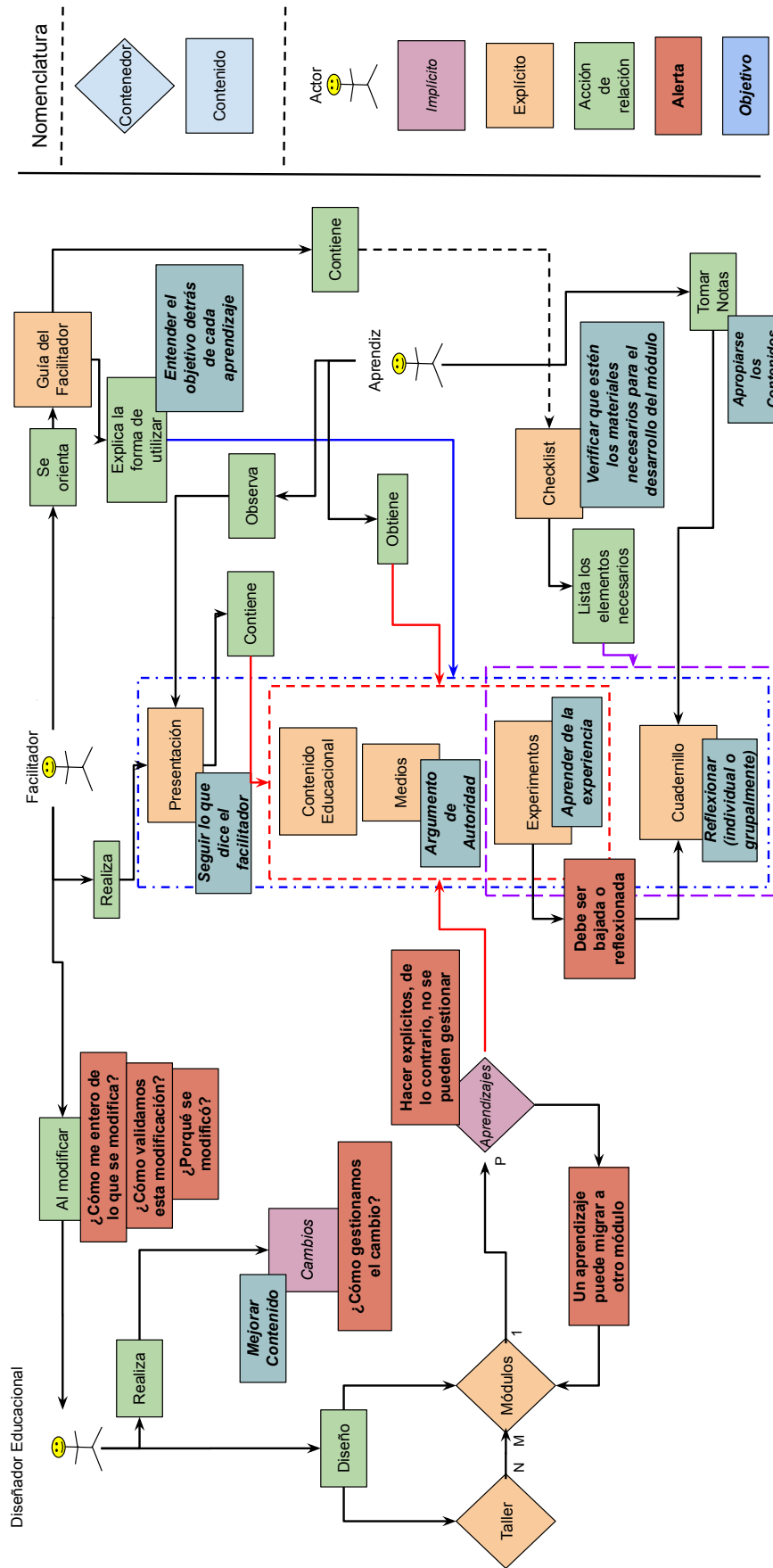


Figura B.1: Diagrama general detallado de la situación actual v20181211