

PROYECTO FINAL

Informe de Proyecto Final de Carrera Aplicación de Soporte para el ENIEF 2019

CARRERA: Ingeniería en Sistemas de Información

DIRECTOR: Martín Domínguez

REALIZADORES: Olivencia, Ramiro

Albertengo, Federico

Tabla de contenido

Agradecimientos	7
Ramiro.....	7
Federico	7
Introducción	8
Fundamentación	9
Delimitación y dominio del problema.....	11
Objetivos generales	11
Objetivos específicos	11
Población	11
Ubicación en el tiempo y el espacio	12
Áreas de conocimiento.....	12
Proceso de Desarrollo.....	13
¿Qué es Kanban?	14
Principios y Prácticas	15
Visualizar el trabajo.....	16
Limitar la cantidad de trabajo en progreso	16
Lograr que el trabajo fluya mejor a través del proceso	17
Kanban y Scrum: una comparación.....	18
Kanban en práctica.....	19
Puesta a punto.....	19
Ciclo de vida	20
Adaptando el proceso	21
Implementación del Proceso de Desarrollo	24
Motivo de la elección de Kanban	24
Detalles de la Implementación	25
Prácticas	28
Story Splitting.....	28
Reuniones con clientes	29
Refinamiento de historias.....	29
Pruebas de aceptación	29
Pruebas unitarias	29
Code review.....	29
Convenciones de nombres	29
Definición de Hecho	31

Tecnologías y Herramientas de Software.....	32
Desarrollo del Cliente	32
Tecnologías	32
Herramientas	32
Librerías y componentes de terceros	32
Desarrollo del Servidor	33
Microservicios	33
JSON Web Tokens (JWT).....	33
Tecnologías	35
Herramientas	35
Pruebas.....	35
Tipos de pruebas	35
Herramientas	38
Gestión del Proyecto	39
Herramientas	39
Desarrollo del sistema.....	40
Listado de módulos desarrollados	40
Inicio de Sesión.....	40
Mi Cronograma	40
Próxima Actividad	40
Actividades Favoritas.....	40
Asistencia	41
Disertantes.....	41
Localización	41
Ajustes.....	41
Sponsors.....	41
Perfil	41
Acerca de.....	41
Ambiente de Desarrollo	41
Local.....	42
Staging o Pre Producción	42
Producción.....	42
Prototipos previos.....	43
Prototipo base.....	43
Incrementos previos.....	43

Arquitectura.....	45
Cliente	45
Servidor	45
Deployment.....	46
Diagrama de arquitectura.....	46
Requerimientos del incremento	47
Requerimientos funcionales	47
Requerimientos no funcionales	48
Modelado	49
Modelo de clases	49
Diagramas de flujo	51
Diagramas de secuencia.....	53
Diagrama de despliegue	56
Diagrama de entidad-relación	56
Actividades y tareas	58
Gestión del Proyecto.....	60
Framework Essence.....	61
Alphas definidos por el SEMAT Kernel	62
Utilización del SEMAT Kernel para la gestión del proyecto.....	77
Iteración 0	79
Iteración 1	106
Iteración 2	126
Iteración 3	144
Iteración 4	159
Iteración 5	177
Retrospectiva final	191
Conclusión.....	198
Aportes del proyecto.....	198
Utilización de la aplicación en el evento	199
Experiencia del proyecto	200
Puesta en práctica de un proceso ágil de desarrollo basado en una framework nuevo para el equipo	200
Utilización del framework Essence y del SEMAT Kernel para la gestión del proyecto	200
Desarrollo de aplicaciones móviles híbridas con Ionic.....	200
Experiencia del equipo colaborando en el desarrollo de iniciativas previas.....	200
Evaluaciones de riesgos detallados en un proyecto de software.....	201

Glosario	202
Ad hoc	202
Alpha	202
Angular	202
API RESTful.....	202
Aplicación móvil híbrida	202
Backend.....	202
Cadencia.....	202
Chair	202
Code review.....	202
Código QR	203
Conferencia	203
Conferentia	203
Congreso	203
Control de versión.....	203
Cordova.....	203
CSS3	203
Cuello de botella	203
Deploy	203
Depuración	203
Despapelización.....	203
Diseño responsive	204
ENIEF 2019	204
Fast track.....	204
Feedback.....	204
Flujo de información.....	204
Framework.....	204
Funcionalidad	204
Gestión integral.....	204
Grado de personalización	204
HTML 5.....	204
HTTP	205
IDE.....	205
IETF.....	205
Implementación.....	205

Incremento.....	205
Ionic	205
Jasmine	205
Javascript.....	205
JSON	205
JWE	205
JWS.....	205
JWT	206
Kanban	206
Karma.....	206
Lean Thinking	206
Librería	206
Look & feel.....	206
Métrica.....	206
Microservicio.....	206
Notificación push.....	206
npm.....	206
Ofimática.....	207
Panelista	207
Performance	207
PFC	207
PHP	207
Plataforma de software	207
Plugin.....	207
Prototipo	207
Regresión	207
Requerimiento	207
Retroalimentación	207
RFC	208
SCRUM	208
SCSS.....	208
Sesión.....	208
Simposio.....	208
SOAP.....	208
Stage	208

Suite de testing	208
Tech stack	208
Teoría de las limitaciones (Theory of constraints)	208
Test runner	208
Token.....	209
Tokenización.....	209
Trello.....	209
TypeScript.....	209
Unit testing.....	209
UTF-8	209
Way of working	209
WebStorm.....	209
Bibliografía y referencias.....	210
Libros	210
Papers.....	210
Artículos	210

|

Agradecimientos

Ramiro

A Lisandro Vrancken, Martín Domínguez y la cátedra de Proyecto Final por ser parte vital para poder llevar este proyecto a buen término.

A Pablo Kler por la dedicada asistencia y las incontables reuniones a lo largo de este proyecto.

A Claudio Bracalenti por la inspiración para profundizar en la filosofía de los sistemas.

A Rafael Olivencia, por la chispa inicial.

A Viviana, Oscar y Valentina por el apoyo de siempre.

A Analía, por la paciencia y los valiosos aportes.

Federico

En primer lugar, quiero agradecer a mi tutor Martín Dominguez y al jefe de trabajos prácticos Lisandro Vrancken por los conocimientos y apoyo que me sirvieron de guía durante las etapas de este proyecto, resolviendo todas las dudas para poder alcanzar los resultados deseados.

También quiero agradecer a Pablo A. Kler, representante de los stakeholders, así como a todos los organizadores del ENIEF 2019 por brindarme la oportunidad y los recursos y herramientas necesarios para la realización de este proyecto.

Además, quiero brindar agradecimiento a los organizadores de los eventos en los que se utilizó el prototipo, los cuales me dieron la oportunidad de realizar pruebas de la plataforma en un contexto real, así como darnos feedback sobre la misma.

Por último, quiero agradecer a mi familia, amigos y compañeros por el apoyo emocional y la contención que me ofrecieron durante todo este proceso.

Introducción

El presente Proyecto Final de Carrera (PFC) consiste en el desarrollo de una aplicación móvil orientada a permitir la gestión digital integral de un congreso académico.

Un evento de estas características consta normalmente de múltiples actividades simultáneas, en diferentes lugares geográficos, las cuales requieren estar diferenciadas según áreas temáticas, horarios y sesiones. A su vez, los organizadores de este tipo de eventos suelen brindar información de interés a fin de facilitar la movilidad y fomentar el turismo dentro de la localidad en cuestión.

Tomando como base un prototipo funcional preexistente desarrollado para presentar la idea frente a los destinatarios del producto, el presente informe refleja el desarrollo de una implementación completa de los requerimientos expresados por el cliente y que forman parte del presente documento.

En forma resumida, el presente informe de PFC detalla el desarrollo de un sistema de información consistente en la aplicación móvil de gestión del evento ENIEF 2019, la cual representa un incremento de desarrollo respecto de un prototipo de plataforma de software preexistente.

Fundamentación

En las diferentes áreas del saber, se ofrece la posibilidad de asistir a eventos académicos que permiten la adquisición de nuevos conocimientos. Un evento académico consiste en un conjunto de actividades que implican algún tipo de enseñanza o capacitación. Entre estos se pueden incluir congresos, simposios, jornadas y conferencias.

Para su gestión, existen actualmente diversas herramientas informáticas disponibles en el mercado. Estas facilitan la promoción, comunicación y acceso a la información de un evento en particular. Como ejemplo se pueden nombrar Eventbrite, Billetto y Facebook Events.

Sin embargo, en ciertas ocasiones, lo puesto a disposición por estas no es capaz de satisfacer los requerimientos del usuario. Esto se debe a dos motivos principales. En primer lugar, cada aplicación puede resultar incompleta y no ofrecer todas las funcionalidades necesarias. En segundo lugar, la flexibilidad y el grado de personalización ofrecidos por estas puede no ser suficiente. De ser así, los organizadores del evento no tienen otra alternativa más que solicitar el desarrollo de una aplicación propia, lo cual supone incurrir en altos costos y en extensos tiempos.

Por este motivo, se pensó que resultaría de gran utilidad el desarrollo de una plataforma modular y personalizable para la generación de aplicaciones independientes que permitan la gestión integral de eventos académicos. Dicha plataforma debía brindar la posibilidad de incorporar las siguientes funcionalidades:

- permitir la validación de la asistencia de los participantes a las distintas actividades ofrecidas, con la capacidad de generar reportes de asistencia;
- poner a disposición información turística y de movilidad a los asistentes, siendo que generalmente estos provienen de ubicaciones geográficas diversas;
- ofrecer información detallada acerca de los disertantes y participantes de las diversas actividades, pudiendo incluir a chairs, directores de sesiones y panelistas;
- llevar a cabo encuestas a fin de conocer la opinión de los asistentes respecto de las actividades desarrolladas;
- generar cronogramas personalizados para cada asistente, especialmente útil en los casos en los que existen actividades en simultáneo;
- permitir la interacción entre los asistentes y los organizadores vía digital por medio de la plataforma;
- brindar la posibilidad de incorporar publicidad de sponsors y patrocinadores.

De esta forma, el usuario sería capaz de obtener una aplicación totalmente adaptada a su evento y que incluya todos los elementos necesarios.

En el presente Proyecto Final de Carrera se detallan los requerimientos para el desarrollo de dicha plataforma así como también la generación de una aplicación para un evento concreto, a saber, el XXIV Congreso sobre Métodos Numéricos y sus Aplicaciones - ENIEF 2019.

Se seleccionó este evento en particular por solicitud de los propios organizadores. Estos, debido a la escala de asistentes y del flujo de información a manejar, determinaron la

necesidad de adquirir una herramienta informática que permita llevar a cabo una gestión integral.

Por otra parte, es importante destacar que previo a este proyecto, los mismos desarrolladores, ya habían construido un prototipo de la plataforma y generado aplicaciones para otros eventos similares, motivo por el cual los organizadores del ENIEF decidieron contactarlos.

Delimitación y dominio del problema

El presente desarrollo se sitúa dentro del dominio de la organización de conferencias académicas, tomando la palabra “conferencias” como un término general que puede describir desde grandes congresos a jornadas de un solo día.

Las conferencias académicas tienen como fin la discusión, difusión y el intercambio de conocimientos por parte de personas dentro de una determinada especialidad.

Estas, normalmente, están compuestas por una lista de actividades y se rigen por un cronograma que detalla cuándo y dónde se realizan dichas actividades. Toman, así, la forma de eventos dentro de eventos, razón por la cual, las plataformas de uso masivo existentes no cuentan con el nivel de personalización requerido.

En el caso aquí presentado, el incremento del sistema de información desarrollado se orientó a servir de soporte tecnológico para el ENIEF 2019, congreso académico organizado en la ciudad de Santa Fe, el cual versa sobre métodos numéricos y sus aplicaciones.

A su vez, además de esta aplicación concreta, se creó una plataforma para la generación de aplicaciones móviles para este tipo de eventos. Se bautizó este proyecto de plataforma de software con el nombre de “*Conferentia*”.

Objetivos generales

El proyecto tiene como objetivo general el desarrollo de una aplicación móvil que permita servir de soporte tecnológico a la organización y gestión de un congreso académico, el cual consta de múltiples actividades dentro de su cronograma.

Objetivos específicos

- Desarrollar una aplicación móvil que permita el soporte tecnológico para la organización y gestión, del ENIEF edición 2019 - XXIV Congreso sobre Métodos Numéricos y sus Aplicaciones, a desarrollarse entre el 5 y el 7 de noviembre de 2019
- Utilizar el prototipo como plataforma base para la generación de aplicaciones móviles para otros eventos.
- Analizar la viabilidad de Ionic Framework como plataforma para el desarrollo de aplicaciones móviles híbridas comerciales, facilitando la implementación en diferentes dispositivos y sistemas operativos.

Población

La población del proyecto está conformada por el equipo de desarrolladores, un equipo de consultores principales (2 personas), que son parte de la organización del evento, el resto de los organizadores del evento (30 personas), que la utilizarán para dar información y tomar asistencia y, por último, los asistentes del congreso (estimado en 400 personas). Además de

los involucrados de forma directa, existen sponsors del evento vinculados indirectamente gracias a publicidades existentes dentro de la aplicación.

Ubicación en el tiempo y el espacio

El proyecto fue desarrollado en la ciudad de Santa Fe desde fines de mayo de 2019 hasta principios de agosto de 2019, con un posterior lanzamiento durante el mes de noviembre de 2019, como fecha límite interna para entregar el producto al cliente.

Áreas de conocimiento

Algunas de las áreas de conocimiento involucradas en este proyecto son:

- desarrollo de software;
- algoritmos y estructuras de datos;
- aplicaciones móviles híbridas;
- análisis de requerimientos;
- diseño y administración de base de datos;
- ingeniería de software;
- arquitecturas de software.

Algunas de las asignaturas relacionadas con este proyecto son:

- gestión de datos;
- administración de base de datos;
- métodos ágiles;
- desarrollo de aplicaciones móviles;
- paradigmas de programación;
- diseño de sistemas;
- ingeniería de software;
- diseño de software basado en arquitecturas.

Proceso de Desarrollo

Para concretar el presente proyecto, el grupo decidió hacer uso del marco de gestión de tareas Kanban como soporte a la implementación de un proceso de desarrollo basado en la filosofía ágil. Para el presente PFC se llevó adelante una formalización del proceso de desarrollo heredado de las etapas anteriores, las cuales competen al desarrollo de los prototipos preliminares anteriormente mencionados. Estos prototipos fueron desarrollados mediante incrementos en las funcionalidades y características deseadas por los clientes de las diversas aplicaciones que el equipo desarrolló previamente a abordar el incremento de software que da lugar al presente proyecto.

Los procesos ágiles son especialmente apropiados para desarrollos incrementales, debido a que ofrecen la capacidad de continuamente adaptarse a los cambios. La razón de ello es que no establecen un plan general al inicio del proyecto, sino que el plan se divide en pequeños planes realizables a lo largo de pocas semanas. De esta manera la gestión del proyecto en desarrollo se torna considerablemente más manejable y clara, a medida que los requerimientos se consolidan y los riesgos - conjuntamente con las medidas para mitigarlos - comienzan a poder ser identificados con más facilidad.

Uno de los factores fundamentales que se tuvieron en cuenta para la elección del proceso de desarrollo, con su correspondiente metodología, fue la necesidad de una adecuada gestión del riesgo. Esto se debe a que la duración del evento académico al que la aplicación apuntó a brindar soporte no excede los cinco días. Por este motivo, la aparición de un error no contemplado y la imposibilidad de solucionarlo a tiempo hubiese provocado la total inutilidad del producto desarrollado. A esto se suma el hecho de que los organizadores decidieron reemplazar completamente la gestión en formato físico por la gestión digital a través de la aplicación, por lo que no existe un respaldo material en caso de falla.

Por otra parte, se debe tener en cuenta que se parte de un prototipo, el cual ya había sido validado y probado en instancias previas. Dichas instancias refieren a eventos de menor escala y/o menor tenor de requerimientos. Esto posibilitó, además de validar funcionalidad, evaluar cuestiones referidas a la performance de la plataforma. Estas cuestiones tuvieron un fuerte impacto a la hora de determinar las prácticas a utilizar. A su vez, fueron de suma importancia para lo que respecta a la gestión del riesgo, ya que permitió realizar pruebas del producto a gran escala y en situaciones similares a las necesarias.

Se presentan a continuación una serie de secciones que tienen como finalidad la definición del proceso de desarrollo elegido, para luego pasar a brindar al lector una perspectiva amplia acerca de qué es Kanban, en general, y cómo llevó adelante el equipo de trabajo su implementación, en particular. Se detalla el origen de Kanban (*¿Qué es Kanban?*), sus principios y prácticas asociadas, el marco teórico de su funcionamiento dentro de un proceso de desarrollo de software, los detalles de la implementación de *Kanban* del grupo y luego una exposición de los fundamentos por los cuales se decidió elegirlo. Finalmente, se procede a detallar la *definición de hecho* establecida por el equipo a lo largo del desarrollo, mediante la cual se evaluó a cada una de las tareas que el equipo llevó adelante durante el proceso.

¿Qué es Kanban?

Kanban es un marco de gestión de tareas que resulta aplicable al desarrollo de software y que es, actualmente, utilizado como soporte de una cantidad significativa de metodologías basadas en el *desarrollo de software ágil*¹. Es, conjuntamente con Scrum, uno de los marcos de gestión de tareas, también llamados *frameworks*, más populares para el desarrollo de software.

El framework Kanban está basado en ideas sencillas, pero que poseen la virtud de ser sólidas a la vez que ofrecen flexibilidad gracias a las *extensiones* que pueden generarse desde estas ideas, a las que nos referiremos como *principios*. Estos principios, junto con sus prácticas asociadas, brindan la posibilidad de adaptar el *way of working*² del equipo de desarrollo. La premisa general de Kanban es la de emparejar la cantidad de trabajo en curso con la capacidad de trabajo del equipo.

En su esencia, Kanban se inspira originalmente en los sistemas *kanban* de scheduling utilizados por Toyota para la producción de bienes manufacturados según el formato *just-in-time*. Bajo este formato la producción de unidades está basada en la demanda de los compradores, contrariamente al enfoque estándar bajo el cual se produce un conjunto de bienes y luego se los procede a introducir al mercado.

En su concepción original, un *kanban* involucra una señal desde un agente dentro de un flujo de trabajo hacia otro agente que se encuentra en una etapa previa dentro del flujo. Esta señal transmite al agente previo dos cuestiones:

1. Le notifica que el agente que le sigue ha completado una operación. Esta operación consume un input que el agente que le precede produce.
2. Por tanto, el agente receptor/predecesor debe producir un input de reemplazo para que el agente que le sucede pueda consumir. De esa forma, permite que la producción continúe.

En su adaptación a la industria del software, y dado que la generación de productos es en este caso de *artefactos intangibles*, a lo largo del ciclo de desarrollo los requerimientos están representados por unidades de trabajo reducibles a tareas. Estas tareas son luego asignadas a las personas que forman parte de un proyecto en desarrollo durante una serie de estados que representan el nivel de realización de cada una.

Kanban con una “K” mayúscula se utiliza cuando hablamos de la implementación como *framework* o marco de gestión de trabajo de desarrollo de software. En su acepción original se utiliza sin capitalizar, como kanban, denotando el sistema de producción original mencionado.

¹ Beck, K. M. & Beedle, M. (2001). *Manifesto for Agile Software Development*. Ward Cunningham.

² Forma de trabajar. Se incluye el anglicismo dado que no hay traducción semánticamente correcta.

Las implementaciones de Kanban, a nivel visual y por razones que se detallarán en secciones subsiguientes, usualmente se representan con un tablero como el de la figura 1. Este tablero está dividido en columnas que representan los diferentes estados e incluyendo las tareas como tarjetas o post-its que se mueven, de forma horizontal, entre los distintos estados. Puede observarse un tablero Kanban sencillo. Mencionamos también que el término *kanban*, de origen japonés, se traduce a *tablero*³ en idioma castellano.

Los estados que usualmente pueden encontrarse en un tablero de Kanban básico son:

1. Backlog – ítems esperando a entrar en cola de procesamiento,
2. Por hacer / To Do – ítems en cola esperando a ser procesados,
3. En proceso / Ongoing – ítems que están siendo procesados,
4. Hecho / Done – ítems que han sido completados y terminaron el proceso.



Figura 1 – Tablero simple de Kanban⁴. Pueden observarse las columnas representando los estados antes mencionados a través del proceso y post-its representando cada uno a una tarea.

Principios y Prácticas

Kanban está orientado a hacer que el flujo de trabajo o *workflow* sea más veloz a través de toda la cadena de valor; se intenta de reducir al mínimo posible el tiempo desde el momento en el que se genera una idea, un concepto o un requerimiento hasta el momento en el cual un artefacto de software pasa a la etapa de despliegue en el ambiente de producción.

Para este objetivo, Kanban utiliza una serie de principios. El equipo de trabajo debe respetar estos principios y tomarlos como consigna para posibilitar un proceso de desarrollo controlable y mesurable. Utilizados como filosofía práctica para gestionar el trabajo, estos principios permiten cumplir el objetivo de minimizar los tiempos de procesamiento de las tareas dentro del proceso.

³ El término *signaling board* se utiliza en Hammarberg et al (2014). Lo tradujimos aquí como *tablero*.

⁴ **Tablero simple de kanban [Figura].** Recuperado de <https://d112uwirao0vo9.cloudfront.net/wp-content/uploads/2018/11/Simple-Kanban-Board-1024x628.png>

Dependiendo de la bibliografía consultada, los principios base de Kanban varían, o bien se amplía la definición a principios y prácticas base. Ciñéndonos a la definición de Hammarberg y Sundén (2014)⁵, mencionaremos tres principios base. Estos principios son:

- visualizar el trabajo y las políticas de trabajo,
- limitar la cantidad de trabajo en progreso,
- lograr que el trabajo fluya mejor a través del proceso.

Estos principios, aplicados de forma continua y regular, permiten lograr un flujo de trabajo ágil y fluido. En tope de los principios, luego las prácticas de cada equipo de trabajo, las cuales resultan extensibles y modificables, son las que brindan la flexibilidad y adaptabilidad de Kanban a cada proyecto y proceso de desarrollo. Procederemos a describir, brevemente, la naturaleza de cada uno de estos principios.

Visualizar el trabajo

Visualizar el trabajo significa poder brindar una representación visible a nuestro modelo de flujo de trabajo. En general existe una variedad de maneras de lograrlo, siendo por defecto para Kanban la de utilizar un tablero como el de la Figura 1. En este tablero, una serie de columnas representa el progreso a través de distintos estados dentro de un flujo de trabajo.

Kanban aboga por la visualización del trabajo y de las políticas de trabajo por dos razones:

- distintas personas dentro de la organización pueden tener una comprensión diferente acerca de qué es lo que representa el flujo de trabajo,
- visualizar la información ayuda a poder transmitirla de manera más efectiva y condensada.

LIMITAR LA CANTIDAD DE TRABAJO EN PROGRESO

El segundo principio se refiere a limitar la cantidad de trabajo en progreso o *work in progress (WIP)*. La manera más sencilla de lograrlo es la de implementar restricciones en la cantidad de ítems de trabajo o tareas que se encuentran dentro de un mismo estado, o columna del tablero, a la vez. Esto permite la reducción de cuellos de botella, limitando la cantidad de tareas en simultáneo. Este principio de Kanban se basa en que cuantos más elementos de trabajo haya en curso, más se cambia el contexto, lo cual contribuye a entorpecer el camino para finalizar el trabajo en curso.

Según el principio 1, esto significa que debemos hacer visible esta información y que debe ser parte del tablero de Kanban. Una manera simple de lograrlo es señalar el número de ítems máximos por columna, en tope de cada columna junto a la descripción.

Una ventaja importante de Kanban es que se pueden modificar las prioridades sin afectar al equipo de desarrollo. Esto es así debido a que el equipo trabaja enfocadamente en lo que se está en progreso en cada momento y no en lo que se tiene que hacer posteriormente.

⁵ Hammerberg, M. & Sundén, J. (2014). *Kanban in Action*. Manning Publications, 1st edition.

En la imagen adjuntada a continuación podemos ver una aplicación del segundo principio, a través de un tablero de 5 estados.

Ready for Design	Design	Ready for Build	Build	Ready for Deployment
Item 7	Item 5	Item 3	Item 2	Item 1
Item 8	Item 6	Item 4		
Item 9				

Figura 2. Tablero de Kanban con 5 estados, incluyendo el límite de ítems permitidos en cada uno. Los estados refieren a *listo para diseño*, *diseño*, *listo para construcción*, *construcción* y *listo para despliegue*, respectivamente de izquierda a derecha:

Lograr que el trabajo fluya mejor a través del proceso

Es este tercer principio el cual, de forma implícita, otorga la posibilidad de que el equipo pueda mejorar su *way of working*, y es ahí donde el segundo elemento que da forma a Kanban entra en juego: las prácticas. Si bien Kanban por se no prescribe reglas o prácticas concretas, la definición y puesta en uso de estas mediante el consenso de los equipos de trabajo es una de las maneras en las cuales podemos mejorar nuestro proceso.

En un proceso de desarrollo, nuestro *way of working* siempre podrá ser mejorado, dado que siempre existirán cuellos de botella que nos llevarán a bloqueos o ralentizaciones a través del proceso. La buena noticia es que la existencia de estos problemas se nos revelará de forma visual, gracias al principio 1, y por norma general, es de esperar que los grandes problemas a resolver dentro del flujo de trabajo sean, por esta misma razón, los que detectemos primero.

Es frente a las situaciones donde se generan cuellos de botella o bloqueos cuando tenemos la posibilidad de mejorar. Y es ahí donde podemos aprovechar para comenzar a establecer prácticas dentro de nuestro proceso para hacer que el trabajo fluya mejor a través de él.

Una forma de realizar una regular identificación de estos aspectos a mejorar consiste en establecer la práctica de generar reuniones, muchas veces a intervalos regulares, que nos permitan generar colaboración entre miembros del equipo en la detección y propuestas de mejora. Esta práctica toma el nombre de reuniones de retrospectivas, o *retrospectives* en el inglés original. Son reuniones del equipo de trabajo dedicadas a analizar qué fue bien, qué fue mal y qué deberíamos cambiar para mejorar el proceso de trabajo desde la reunión anterior.

Se detallará en secciones subsiguientes ejemplos acerca de cómo un equipo en general lleva adelante una implementación de Kanban y cómo el equipo que desarrolló el presente Proyecto Final de Carrera implementó el propio, mediante la adopción de prácticas tendientes a mejorar su flujo de trabajo.

Kanban y Scrum: una comparación

En comparación con Scrum, las características que definen a Kanban y que hacen a sus diferencias se mencionan a continuación:

	SCRUM	Kanban
Cadencia	Sprints de longitud fija periódicos (por ejemplo, dos semanas)	Flujo continuo
Metodología de publicación	Al final de cada sprint, si lo aprueba el propietario del producto	Entrega continua o a discreción del equipo
Funciones	Propietario del producto, experto en scrum, equipo de desarrollo	No existen funciones. Algunos equipos cuentan con la ayuda de un orientador ágil
Métricas clave	Velocidad	Tiempo del ciclo
Cambio de filosofía	Los equipos deben evitar cambios en la previsión durante el sprint. De lo contrario, se sacrifica el aprendizaje sobre la estimación.	Los cambios pueden suceder en cualquier momento.

Tabla 1 – Diferencia entre Scrum y Kanban

Kanban en práctica

Esta sección tiene la finalidad de ejemplificar cómo, de forma general, un equipo de trabajo genera una implementación de Kanban siguiendo los principios enunciados anteriormente y realiza paulatinamente ajustes para mejorar su proceso. La palabra implementación, al igual que en el caso del uso que se le da en la jerga del desarrollo de software, refiere a la adopción e integración de una herramienta dentro de un flujo de trabajo.

A fin de ejemplificar cómo funciona Kanban, procederemos a comentar cómo se comienza la implementación de un proceso basado en Kanban, para luego ejemplificar el ciclo de vida de un ítem de trabajo a través del proceso en la sección “Ciclo de vida” y por último dar un ejemplo de la aplicación del principio 3 en la mejora del proceso en la sección “Adaptando el proceso”.

Puesta a punto

Para poner en marcha un proceso basado en Kanban, debe cumplirse con algunos elementos que sirven de sostén. Es aquí donde se definen los soportes que nos permitirán aplicar los principios 1 y 2 que Kanban prescribe y la introspección que se realizará sobre el *way of working* que da lugar al principio 3. Se detalla aquí lo necesario para echar a andar un proceso basado en Kanban con el mínimo necesario de requisitos para que funcione.

Como parte del principio 1:

- Se requiere de un soporte, sea físico o digital, para acomodar los ítems de trabajo.
- Se debe proveer con una forma de identificar a cada uno de los ítems de trabajo a realizar
- Se debe contar con un mecanismo para visibilizar a qué persona del equipo de desarrollo se encuentra asignado cada uno de estos ítems de trabajo.
- Se debe proveer una manera de establecer cuáles serán los distintos estados entre que un ítem está especificado como ítem de trabajo a realizar hasta el estadio en el cual se lo determina como terminado.
- Se debe convenir una definición de terminado, consensuada por el equipo de trabajo.

Como parte del principio 2:

- Se debe convenir cuál es la cantidad máxima de ítems de trabajo que, en simultáneo, se encuentran dentro de cada uno de los distintos estados de realización parcial. Para cumplir con el principio 1, además, esta información debe ser visible para todo el equipo.

Como parte del principio 3:

- Se debe convenir cuál será la manera con la cual el equipo procederá a inspeccionar cómo se encuentra trabajando y cómo evalúa cuáles son los problemas que

obstaculizan el fluir de los ítems de trabajo a través del proceso. Usualmente se opta por reuniones, en forma de ceremonias regulares, para lograrlo.

Respecto de estas precisiones, se incluyen algunas aclaraciones extra relevantes a estos requisitos, a fin de diferenciar cuestiones que son necesarias de establecer de una forma determinada y otras que dan margen para generar decisiones según lo convenido por el equipo de trabajo.

- Aunque se intenta aquí establecer una relación de linealidad respecto de los puntos requeridos para que una implementación cumpla con cada uno de los principios, algunos de estos requerimientos pueden tener relación con más de un principio, según los básicos establecidos y los que más tarde el equipo determine.
- Si bien el tablero o dashboard no está prescripto por Kanban ni es indispensable contar con uno para llevar adelante el proceso, se lo utiliza debido a que es una de las maneras en las que se logra otorgar visibilidad y así cumplir con el principio 1.
- Existen marcos de gestión y metodologías, tales como Scrum y XP respectivamente, que tampoco prescriben la utilización de un tablero. Kanban utiliza el tablero a manera de *estándar de facto* debido a razones históricas y de origen, lo cual se relata en la sección “*¿Qué es Kanban?*”.
- El mínimo de estados entre los que se debe diferenciar el ciclo de vida de un proyecto es de cuatro: trabajo pendiente, trabajo por hacer, trabajo en realización y trabajo terminado. Un buen reflejo de este mínimo puede observarse en la figura 1.
- Para representar ítems de trabajo en Kanban, usualmente se opta por los post-its que reflejen en formato escrito el trabajo a realizar o bien mediante tarjetas digitales en el caso virtual, siendo éstas abstracciones de un post-it del mundo real.
- Si bien las reuniones *daily* y las semanales o quincenales en forma de *retrospectiva* son una manera de proveer un mecanismo de aplicación del principio 3, Kanban no prescribe cada cuánto deben ser realizadas estas reuniones ni su duración.

Ciclo de vida

Como ya hemos mencionado, cada ítem en un tablero de Kanban representa una pieza de trabajo a ser realizada. Reflejando las diversas etapas en las cuales puede encontrarse un trabajo, Kanban utiliza estados para identificar qué tan completo se encuentra un ítem de trabajo que existe dentro del proceso.

En la figura 1 pueden observarse seis elementos, ubicados a lo largo de las diferentes columnas del tablero, nombrados por letras de la A a la F. El ítem A, que se encuentra en la columna, “Done”, se encuentra aquí debido a que su estado determina que ya el trabajo que representa ya ha sido realizado y terminado. Para llegar allí, sin embargo, ha tenido que atravesar todos los demás estados que se encuentran representados en el tablero.

Cuando se determina que hay un determinado trabajo a realizar dentro de las actividades de desarrollo, encontrándose ese trabajo de valor para el producto en desarrollo, se lo representa como uno o más ítems de trabajo. Cada ítem contiene una descripción de qué es lo que debe ser realizado, en forma de tarea assignable a un integrante del equipo de desarrollo. Una vez explicitado el trabajo de esta manera, se procede a agregarlo al trabajo futuro a realizar,

asignándole el estado de “agregado al backlog”; representado en el tablero por la columna que se encuentra más a la izquierda.

Una vez que se asigna un ítem de trabajo a un integrante del equipo, el ítem procede a pasar al estado de “a realizar” (*to-do*). Recordando el principio de limitar la cantidad de trabajo en progreso, en esta columna y las que se encuentran a la derecha, con excepción de “Done”, siempre tendrán un número máximo de elementos, determinados por la cantidad múltiplo de los miembros del equipo y la cantidad de ítems máximos asignables a cada uno de ellos (*miembros x ítems máximos asignables*).

Cuando un integrante del equipo de trabajo procede a tomar un elemento de la lista de *to-do*, comienza a trabajar en él y produce resultados parciales que llevan al ítem a acercarse al objetivo definido, el ítem se encontrará en el estado “en realización” (columna *ongoing*). Se encontrará allí en tanto y en cuanto el ítem aún no se determine como completado y mientras que el ítem pueda ser susceptible de ser trabajado, cuestión en la que ahondaremos luego.

Una vez que el ítem se determina como terminado, completo, se procede a asignarle el estado “done”. Dentro de Kanban, esto representa que un ítem está listo para ser parte del producto en desarrollo. La definición de cuándo un ítem de trabajo se encuentra terminado dependerá siempre de la implementación particular de Kanban que un equipo de trabajo realice, no siendo siempre extrapolable a otros equipos de trabajo. Procederemos a detallar más sobre esta cuestión en la sección siguiente.

Adaptando el proceso

Example of a Kanban Board					
Backlog	In Progress (3)	Peer Review (3)	In Test (1)	Done	Blocked
					
Fast Track/ Defect					

Figura 3 – Tablero de Kanban⁶. En él pueden observarse columnas adicionales a las de la figura 2 y una sección horizontal para hacer seguimiento de defectos que requieren prioridad.

La aplicación del tercer principio deriva, inevitablemente, en cambios en nuestro proceso y por tanto en nuestra forma de trabajar y de aplicar Kanban. Es en la adaptación del proceso donde se da lugar a la mejora de colaboración entre los miembros del equipo de trabajo y a la evolución experimental del proceso. Se intenta siempre seguir el método científico a la hora de evaluar qué medidas el equipo adopta para mejorar su proceso.⁷

Existe un universo de posibilidades respecto de qué puede cambiarse en un proceso de trabajo para mejorarlo, siendo los cambios y adaptaciones realizados dependiendo del equipo de trabajo, del dominio del proyecto en cuestión y de las reglas de negocio a las que el proyecto debe obedecer.

La figura 3 refleja un tablero de Kanban que incluye no sólo un mayor número de columnas sino una fila adicional, bajo el nombre “Fast Track / Defect”, que se encontraban ausentes en los tableros anteriores. Si bien puede que un equipo experimentado haya tomado la decisión de comenzar con un tablero de estas características al inicio de un proyecto, conociendo cómo debe ser el proceso aplicado, en este caso se invita al lector a pensar este ejemplo como la aplicación paulatina de decisiones respecto de cómo adaptar un su proceso de trabajo para obtener un tablero de estas características.

A partir del tablero se pueden inferir ciertas características del equipo de trabajo al cual pertenece y de las consideraciones del equipo respecto de cómo hacer el trabajo. Por ejemplo, en este caso:

- La definición de *terminado* del equipo requiere de una revisión por pares y una etapa de pruebas para cada ítem como parte de su ciclo de vida, a fin de evaluar el atributo *calidad* como parte de cuándo un ítem se encuentra terminado.
- Existe una columna adicional para almacenar aquellos ítems de trabajo que se encuentran bloqueados - estado que refleja que no puede avanzarse en la concreción de los mismos hasta tanto se cumpla una condición determinada que atañe a otros ítems o a condiciones externas al proyecto.
- Puede observarse una fila de *fast track*, la cual contiene ítems que son prioritarios de revisión por estar relacionados a algún defecto de software, el cual puede ser un *bug* o un problema de *regresión* que debe de solucionarse rápidamente.

Focalizarse en mejorar cómo se aplican los principios 1 y 2 a nuestra implementación de Kanban es una definición acabada y resumida de qué es lo que intenta comunicar el tercer principio.

⁶ **Example of a Kanban Board.** Recuperado de https://en.wikipedia.org/wiki/Kanban_board#/media/File:Kanban_board_example.jpg

⁷ Hammerberg, M. & Sundén, J. (2014). **Kanban in Action**. Manning Publications, 1st edition, p. 102.

Adaptar el proceso también puede implicar otras cuestiones tales como la prescripción de determinadas prácticas por parte del equipo de trabajo que no son parte esencial de Kanban. Los ejemplos que pueden nombrarse son incontables y dependerán siempre de la creatividad y el consenso de los integrantes del equipo de trabajo.

Algunos ejemplos de nuevos principios que se describen en Hammarberg et al (2014)⁸ involucran:

- Hacer explícitas las políticas de proceso, a fin de que las discusiones acerca del proceso encierran evidencia en forma de datos objetivos en vez de que prevalezcan cuestiones anecdóticas o emocionales.
- Implementar ciclos de *feedback*, práctica que permite obtener *feedback* del proceso en forma de retrospectivas del proceso
- Mejorar colaborativamente, evolucionar experimentalmente. Refuerza la práctica de utilizar modelos como la Teoría de Restricciones o el *Lean Thinking* para impulsar al equipo de trabajo en sucesivas mejoras.

⁸ Hammerberg, M. & Sundén, J. (2014). **Kanban in Action**. Manning Publications, 1st edition, p. 100.

Implementación del Proceso de Desarrollo

Motivo de la elección de Kanban

Durante las instancias previas al desarrollo del presente PFC, mientras se llevó adelante el desarrollo de los prototipos e incrementos previos mencionados en el informe, el equipo se encontró ante la situación de tener que elegir un proceso para el desarrollo de estos y una metodología de software para poder organizar el trabajo a realizar y posibilitar la obtención de métricas durante el desarrollo.

Para el desarrollo de los incrementos previos y también del incremento que es parte de este PFC, el equipo de desarrollo hizo uso de un proceso basado en Kanban, con metodología y prácticas ágiles ad-hoc. El equipo se decidió por la adopción de un proceso basado en Kanban por varias razones:

- simpleza de implementación,
- requerimiento de un marco de gestión que posea múltiples recursos bibliográficos, didácticos y prácticos para su implementación,
- ciclos cortos de *feedback* para morigerar riesgos y reducir incertidumbre,
- la experiencia previa de los miembros del equipo de desarrollo en su uso.

Si bien existe una variedad de características asociables a Kanban que resultan atractivas para el desarrollo de software basado en la filosofía ágil, el equipo se centró en estos cuatro puntos para su elección. Posteriormente, se pudo hacer aprovechamiento de características tales como la habilidad de cambiar prioridades mientras se avanzaban en el desarrollo del sistema y la posibilidad de contar con versiones demostrables de los avances realizados a cada momento, pudiendo hacer entregas instantáneas en base a cada una de las características que se determinaban como implementadas y testeadas en el proyecto.

Kanban no prescribe acerca de cuáles son los momentos determinados en los cuales se realiza una demostración o se hace una entrega de un incremento de software al cliente, sino que se asume que el equipo está en condiciones de entregar el software cuando el equipo lo decida o cuando las condiciones dentro del proceso de desarrollo se consideren oportunas. Obedeciendo a esto, el equipo hace uso de iteraciones planificadas para hacer las entregas de los incrementos a lo largo de la ejecución del proyecto.

Las mencionadas iteraciones son las que se encuentran detalladas en la planificación del presente PFC. Cada iteración, a su vez, posee una etapa de planificación, en la cual se toman primeras decisiones acerca de cómo abordar, en alto nivel, las actividades a realizar y se plantean los objetivos de la iteración. Una vez iniciadas las iteraciones, se utiliza Kanban para proceder a gestionar y priorizar el trabajo desarrollado como parte de cada iteración.

La gestión de las iteraciones y del proyecto en sí, como soportes del proceso, se llevó adelante mediante el framework Essence. Se describe en la sección “Gestión del Proyecto” cómo el equipo hizo uso de Essence para la gestión del presente PFC y de las iteraciones de desarrollo del proyecto.

Detalles de la Implementación

Para la implementación de Kanban del presente PFC se hizo uso de un tablero en formato digital, brindado por la herramienta Trello. A lo largo del ciclo de desarrollo se hizo uso en plenitud de las herramientas y extensiones que Trello provee para llevar adelante una implementación óptima de Kanban. La herramienta Trello también fue utilizada durante el desarrollo de los prototipos previos, habiendo sido usada también para generar visibilidad la información de los proyectos de software para los diversos clientes a lo largo de los diferentes desarrollos.

Debido a que consideramos crucial para el éxito del proceso de desarrollo implementado, se optó por no sólo generar visibilidad del tablero para el equipo de desarrollo, sino también para los clientes, a fin de poder visualizar la misma información para todos los interesados a partir de un único lugar en común, teniéndolo como fuente fiable sobre el avance actual del proyecto y el estado de las tareas.

La implementación realizada está fuertemente influenciada en lo expuesto por el libro *Kanban in Action* (2014), de Marcus Hammarberg et al. Este recurso bibliográfico es la base a partir de la cual el equipo formalizó su implementación de Kanban.

Principios

Visibilización del trabajo y las políticas de trabajo:

La visualización del trabajo y de las políticas de trabajo fue implementada gracias a las capacidades brindadas por Trello. Dado que la herramienta específicamente provee las capacidades necesarias para poder implementar un proceso de desarrollo basado en Kanban, esto fue usado a favor para poder cumplir con lo requerido por el primer principio. Las notificaciones y asignaciones de trabajo que fueron utilizadas por los miembros del equipo de desarrollo más la visibilidad por parte del cliente de la información disponible en el tablero también fueron aprovechadas por el equipo para favorecer la visibilidad del proceso.

LIMITAR EL TRABAJO EN PROCESO

A fin de limitar la cantidad de trabajo en proceso, el equipo de desarrollo determinó que el máximo de ítems asignables a cada miembro del equipo que pudiesen estar en las columnas “en proceso” y “en revisión de pares” no podía ser de más de 2 en cada caso. Esto es: cada miembro podía tener a lo sumo 2 ítems en el estado “en proceso” y a lo sumo 2 ítems en el estado “en revisión de pares” simultáneamente y en cualquier momento del proceso de desarrollo.

A lo largo del proceso de desarrollo el número no requirió ser modificado y permitió mantener fluidez a lo largo de la realización del proyecto, con lo cual demostró ser efectivo para la capacidad de trabajo del equipo.

Lograr que el trabajo fluya mejor a través del proceso

Para el tercer principio no se determinaron reuniones específicas a modo de retrospectiva para hacer una evaluación, sino que la incorporación de prácticas se sucedió de manera paulatina según la observación de las necesidades mientras avanzó el proceso de desarrollo.

Los ajustes realizados al proceso surgieron de discusiones en reuniones esporádicas entre el equipo de desarrollo y luego de las evaluaciones hechas por el cliente de las distintas entregas parciales que se realizaron durante el desarrollo del proyecto.

Se determinó como política que cualquiera de los dos miembros del equipo de desarrollo podría especificar ideas a implementar según su criterio, las cuales luego de una evaluación entre pares podría ser agregada como práctica al ciclo de desarrollo. De esta manera se logró ganar experticia a la hora de llevar adelante las diversas prácticas en el ciclo de desarrollo y se determinó la necesidad de especificar nuevos estados adicionales a los iniciales para los ítems de trabajo.

Prácticas

Trello brinda una multitud de características que posibilitan una aplicación sencilla de muchas prácticas de desarrollo, permitiendo incluso la automatización de algunas de ellas y facilitando de gran manera las demás. Trello representa a los ítems dentro del tablero como pequeñas tarjetas, las cuales pueden ser vistas en detalle al clickear sobre ellas. En la vista de detalle pueden agregarse comentarios, generar listas de tareas, adjuntar documentación y enlaces vinculados a la tarjeta y, además, poder visualizar un historial de las modificaciones y transiciones de estado por las cuales ha pasado el ítem de trabajo, facilitando el *tracking* de quién ha realizado cambios, cuáles han sido esos cambios y cuándo los cambios fueron realizados.

Las capacidades de las cuales se hizo uso, ligadas a prácticas adoptadas por el equipo de desarrollo, fueron:

- Estimación del tiempo de desarrollo de los ítems de trabajo
- Item splitting
- Generación de épicas, dada la posibilidad de agrupar ítems de trabajo relacionados en categorías

Estados

Como ya se ha expuesto con anterioridad, Kanban requiere poder identificar visualmente Los estados elegidos para acomodar los ítems a lo largo del proceso de desarrollo fueron variando a lo largo del proyecto, en base a la aplicación del tercer principio de Kanban. Inicialmente, las columnas definidas para agrupar los distintos ítems, según su estado, fueron las siguientes:

- **Backlog**, para agrupar los ítems que requieren un refinamiento en su especificación para, posteriormente, poder transicionar hacia la lista de ítems por hacer.

- **Por hacer**, para agrupar aquellas tareas que ya especificadas en detalle pueden pasar a ser asignadas a uno de los dos miembros del equipo de desarrollo para su realización.
- **En proceso**, para definir qué ítems están actualmente siendo trabajados por los integrantes del equipo de desarrollo.
- **En revisión de pares**, para definir aquellos ítems realizados por un integrante del grupo que requieren validación del otro antes de determinarse como hechos, lo cual incluye revisión de código y pruebas funcionales.
- **Hecho**, para especificar aquellos ítems de trabajo que han pasado

Adicionalmente a las columnas asignadas a los estados, también inicialmente se incluyó una columna adicional, por razones de practicidad:

- **Investigación/Documentación**, para definir tarjetas referidas a documentación y recursos relacionados al proceso de desarrollo de software. Esto incluye recursos de consulta relacionados a las herramientas utilizadas, como así también documentos de texto detallando información recabada de las revisiones entre pares y con el cliente del proyecto.

Posteriormente, mientras avanzó el trabajo, se procedió a añadir tres columnas adicionales a fin de representar dos estados particulares no contemplados anteriormente.

- **Prioritarios**, para representar ítems de trabajo que necesariamente debían tomar precedencia en su realización y compleción antes que otros.
- **Bugs**, para poder especificar ítems de trabajo relacionados a fallas detectadas en el funcionamiento del sistema.
- **Bloqueados**, para especificar ítems de trabajo que no pueden ingresar al flujo de trabajo debido a que hay otro u otros ítems específicos que impiden el desarrollo de éste.

Prácticas

Con el término “prácticas” nos referimos a actividades que son realizadas de forma repetida durante el desarrollo de una iniciativa de software, las cuales brindan soporte al trabajo que un equipo realiza para dar forma a un sistema de software.

Las prácticas generalmente derivan de la metodología o proceso aplicado, aunque los procesos y metodologías ágiles contemplan generalmente la posibilidad de incorporar prácticas no prescriptas por los mismos. Así como se pueden incorporar nuevas prácticas, también se puede decidir no utilizar alguna de ellas según el equipo de desarrollo considere necesario. Para el caso de nuestro proceso de desarrollo, la metodología de desarrollo se genera a partir de las prácticas adoptadas dentro del proceso, siendo la implementación de las prácticas flexible dentro de Kanban debido a la ausencia de prácticas prescriptas por los principios que rigen una implementación de un proceso basado en Kanban.

Para conceptualizar las prácticas y el alcance de las mismas, el equipo utilizó el enfoque del SEMAT Kernel (Ivar Jacobson et al, 2012). Bajo este enfoque, las prácticas se definen como unidades modulares, diferenciadas y separadas, que pueden ser elegidas por un equipo de desarrollo para usar o no usar.

La flexibilidad mencionada sobre las prácticas es proporcionada por su naturaleza de no ser obligatorias, por lo que cada equipo puede elegir cuales desea aplicar y que nivel de esfuerzo dedicarles en su implementación. Se procede, en esta sección, a realizar una lista taxativa de las prácticas de desarrollo que el equipo consideró y posteriormente adoptó como parte de su proceso y que, a su vez, brindaron soporte en la tarea de establecer su metodología de desarrollo. Se incluye en esta sección, a continuación, la lista de las prácticas adoptadas por el equipo para servir de soporte al desarrollo del presente PFC.

Story Splitting

Historias de usuario cortas permiten una más rápida y segura implementación, ya que las pequeñas tareas se mueven a través de cualquier sistema más rápidamente, con menos variabilidad y menor riesgo.

Por lo tanto, dividir historias de usuario grandes en otras más pequeñas es una habilidad obligatoria para cualquier equipo ágil. Pertenece tanto al arte como a la ciencia del desarrollo incremental.

Existen diferentes técnicas para dividir historias de usuario, dentro de las cuales se puede mencionar:

- Pasos del flujo de trabajo
- Variaciones de las reglas de negocio
- Máximo esfuerzo
- Simple/complejo
- Variaciones en los datos
- Métodos de entrada de datos
- Calidades de sistemas diferidos
- Operaciones (Create, Read, Update, Delete [CRUD])
- Escenarios de casos de uso
- Pico de ruptura

Reuniones con clientes

Las reuniones con los clientes es algo muy importante en cualquier desarrollo ágil, pero más en este caso en el que la metodología que se aplicó requiere de un contacto continuo con el cliente de tal forma de poder agregar, eliminar o refinar los distintos requerimientos, así como realizar las pruebas de aceptación.

También se realizaron reuniones con los clientes luego del evento, para obtener un *feedback* para poder mejorar la plataforma.

Refinamiento de historias

El refinamiento de historias de usuario se trata de revisar las tareas futuras para discutir, estimar y establecer un entendimiento inicial del criterio de aceptación. Además de revisar y acomodar las tareas para que se expliquen de la mejor forma y no haya repetidas.

El refinamiento de historias es un proceso continuo que no debe limitarse a un período determinado.

Pruebas de aceptación

La sub-sección [Pruebas de Aceptación](#), dentro de la sección “Tipos de pruebas” (página 39) contiene el detalle de la naturaleza de esta práctica.

Pruebas unitarias

La sub-sección [Pruebas Unitarias](#), dentro de la sección “Tipos de pruebas” (páginas 36-39) contiene el detalle de la naturaleza de esta práctica.

Code review

El code review o revisión de código es la constante revisión del código fuente de un sistema informático con el objetivo de detectar errores tempranamente o detectar alternativas más eficientes. Además, sirve para el aprendizaje y evolución de los desarrolladores, al poder discutir entre ellos las distintas soluciones o la solución ya aplicada.

Convenciones de nombres

A fin de denominar los diversos elementos que componen la base de código del presente proyecto, se decidió, en base a prácticas recomendadas para los lenguajes de programación utilizados y en pos de favorecer la comprensión del código, establecer los siguientes criterios para los distintos elementos, según se detalla en la tabla 4.X que se encuentra a continuación.

Tipo de elemento	Convención de nombres	Ejemplos
Clase	PascalCase	ActividadService, TipoActividad, TipoMarcador, ActividadLink
Atributo	camelCase	descripcion, areaTematica, cronogramaCompleto, actividadesFavoritas

Método	camelCase	getActividades(), getDisertanteById(),
Tablas	snake_case	actividades_asignadas, asistencia

Tabla 2 – Convenciones de nombres

Definición de Hecho

Dentro de los procesos ágiles, la *definición de hecho* es un conjunto de criterios que se considera deben cumplirse respecto de un ítem de trabajo para poder definirlo como completado. Establecer una definición de hecho permite a un equipo contar con parámetros compartidos y comprendidos acerca de la calidad del trabajo realizado, utilizándose para asegurarse de que los despliegues entregados como parte del

La definición de hecho es establecida por el equipo de desarrollo, con posible intervención de los stakeholders y/o líderes de proyecto, y no necesariamente permanece estable a lo largo de toda la ejecución de un proyecto, sino que puede ir siendo mejorada durante el transcurso de éste.

Para la definición de hecho del presente proyecto, el equipo hizo uso de las prácticas que brindan soporte al proceso, las cuales asistieron en la definición de los criterios utilizados. De esta manera, el equipo decidió que para un ítem de trabajo pueda ser determinado como completado, debía cumplirse que:

- El código referente a un determinado ítem de trabajo debe pasar satisfactoriamente por una code review, realizada por los integrante del equipo.
- El código referente a un ítem de trabajo debe pasar satisfactoriamente todas las pruebas unitarias.
- El código referente a un determinado ítem de trabajo debe pasar las pruebas de aceptación.

Tecnologías y Herramientas de Software

Desarrollo del Cliente

Tecnologías

- **Ionic Framework 3.91**, como framework de desarrollo de aplicaciones móviles.
 - Ionic, basado en la tecnología Angular, es un framework que permite desarrollar aplicaciones, web, móviles y de escritorio, de alta performance, con interfaces de usuario adaptables según la plataforma pero compartiendo una base única de código.
 - Ionic, en su versión 3.x, está basado en el framework Angular de Google, el cual utiliza los lenguajes de programación TypeScript, HTML5 y SCSS para desarrollar código web que luego es procesado, compilado y desplegado en varias plataformas a elección.
- **Apache Cordova 7.0.0**, como framework de desarrollo móvil para interactuar con funcionalidades nativas de sistemas operativos y dispositivos.
 - Cordova permite utilizar tecnologías web estándar, como HTML5, CSS3 y Javascript, para el desarrollo de aplicaciones web progresivas y móviles multiplataforma, evitando a los desarrolladores lidiar con los lenguajes nativos de desarrollo de cada una de las plataformas destino de nuestra aplicación.
 - La tecnología fue elegida dado que el equipo de desarrollo tuvo como objetivo el brindar la posibilidad de desplegar la aplicación en las plataformas web, iOS y Android por igual, con una base única de código.

Herramientas

- **WebStorm 2019**, como entorno principal integrado (IDE) de desarrollo.
 - Dado su estatus como uno de los entornos de desarrollo recomendados por la comunidad de desarrollo de Ionic y el conocimiento previo en el uso de este IDE, WebStorm fue la opción predilecta para oficiar de entorno de desarrollo para el código del cliente.
- **Node Package Manager (npm)**, como gestor de librerías y paquetes
 - Dado su estatus predilecto para la gestión de paquetes y librerías de componentes reutilizables

Librerías y componentes de terceros

- **Font Awesome**, como librería de iconos.
 - Font Awesome es una librería que permite incorporar una extensa cantidad de iconos que pueden utilizarse en las vistas HTML.
- **Moment Js**, como librería para el control de fechas y horas.
 - Moment Js es una librería javascript y typescript más conocida y utilizada para el control de fechas en estos lenguajes.
- **Angular QR Code Generator (ng-qrcode)**, como generador de código QR.
 - Elegido por su facilidad para generar un código QR responsive a partir de cualquier dato que se le quiera incorporar.

- **BarcodeScanner Plugin (phonegap-plugin-barcodescanner)**, como plugin lector de código QR.
 - Elegido por ser el plugin preferente a utilizar por la propia documentación de Ionic en la lectura de códigos de barra y QR.
- **OneSignal**, como plugin de manejo de notificaciones push.
 - Seleccionado debido a que es el plugin para notificaciones push multiplataforma recomendado por Ionic en su documentación.

Desarrollo del Servidor

Microservicios

Un diseño basado en microservicios es un enfoque de diseño que permite tener pequeños servicios ejecutándose individualmente, por lo que un problema o un mantenimiento en uno de ellos no afecta al resto.

Los microservicios, además de lo comentado anteriormente, tienen una gran ventaja que es que es fácil de entender ya que la lógica de negocio está bien separada. También, si se da el caso, son más fáciles de migrar o escalar debido a que se pueden tratar individualmente.

En el desarrollo específico para este Proyecto Final de Carrera, la plataforma elegida para los mismos fue PHP, debido a su gran cantidad de servidores que lo pueden manejar en el mercado y por su sencillez en la escritura de código para Backend.

Además de los microservicios, existe un archivo común para todos que contiene funciones y variables referidas a conexiones a la base de datos, consultas, ejecuciones, fechas, entre otras utilidades comunes.

JSON Web Tokens (JWT)

JSON Web Tokens es un estándar abierto basado en JSON para la creación de tokens de acceso que permite la propagación de identidad o privilegios⁹. Estos son utilizados como una forma de autenticación y autorización del cliente de forma cifrada, compacta y que se pueda enviar en los requests HTTP. Los JWT son generados y validados a partir de una clave privada que es almacenada en el servidor.

Citando la definición de JWT en el RFC 7519, JWT es “una cadena de texto que representa un conjunto de solicitudes como un objeto JSON que está codificado en un JWS o JWE, lo que permite que las solicitudes estén firmados digitalmente o en MAC y/o encriptados”.

Se ha decidido utilizar este estándar para la API RESTful desarrollada para este proyecto como método de seguridad, asegurándonos que sólo un usuario registrado y válido puede hacer uso de la misma.

⁹**Wikipedia, La enciclopedia libre (2020), JSON Web Token.** Recuperado el 31 de mayo de 2020, de https://es.wikipedia.org/wiki/JSON_Web_Token

Cuando el usuario inicia sesión y la aplicación llama al servicio correspondiente, se genera un JWT, que es devuelto y almacenado por esta. Cuando se necesita llamar a cualquier otro servicio, la aplicación tiene que enviar al microservicio un JWT válido, de lo contrario el microservicio informará la situación mediante un código de respuesta HTTP 401: Unauthorized con el mensaje de error que indique que no se envió un JWT, el mismo es incorrecto o expiró, en lugar del resultado esperado.

Pasos para validar un JWT

Para la validación de estos JWT, se realizan una serie de pasos, que no necesariamente tiene que seguir un orden si es que no hay relación entre las entradas y salidas de los pasos. Si uno de los pasos falla, el JWT debe ser rechazado.

1. Verificar que el JWT contenga al menos un punto ('.') .
2. Tomar como el encabezado JOSE codificado, la parte del JWT antes del primer punto.
3. Decodificar con base64 el encabezado JOSE codificado, siguiendo la restricción de que no existan saltos de línea, espacios en blanco o cualquier otro carácter adicional.
4. Verificar que la secuencia de octetos resultante es una representación de un objeto JSON válido según la RCF 7159 codificada en UTF-8. Tomar el encabezado JOSE como este objeto JSON.
5. Verificar que el encabezado JOSE resultante incluya sólo parámetros y valores cuya sintaxis y semánticas son ambas entendidas y soportadas o que sea especificado que se las ignore cuando no se las entienda.
6. Determinar si el JWT es un JWS o un JWE usando los métodos que se describen en la RFC 7516 y RFC 7515.
7. Dependiendo de si es JWS o JWE, hay dos casos:
 - Si el JWT es un JWS, se deben seguir los pasos especificados en la RFC 7516 para validar un JWS. El Mensaje será el resultado de decodificar en base64 la carga del JWS.
 - Si el JWT es un JWE, se deben seguir los pasos especificados en la RFC 7515 para validar un JWE. El Mensaje será el texto plano resultante.
8. Si el encabezado JOSE contiene "JWT" como valor de "cty" (tipo de contenido o content type), el Mensaje es un JWT que fue pasado por una serie de operaciones de firmado o encriptado anidadas. En este caso, se debe volver al paso 1 tomando el Mensaje como en JWT.
9. En el otro caso, se debe decodificar el Mensaje utilizando base64 siguiendo la restricción de que no deben utilizarse saltos de línea, espacios u otros caracteres adicionales.
10. Verificar que la secuencia de octetos resultante es una representación de un objeto JSON válido según la RCF 7159 codificada en UTF-8. Tomar a la Solicitud JWT como el JSON Object.

Finalmente, se debe notar que hay una decisión de aplicación sobre cuales algoritmos deben ser utilizados en un contexto determinado. Incluso si un JWT puede ser validado exitosamente, si los algoritmos utilizados en el JWT no son aceptables para la aplicación, se debería rechazar el JWT.

Tecnologías

- PHP 5.6
- MySQL 5.7

Herramientas

- **PhpStorm 2019** como entorno principal integrado (IDE) de desarrollo.
 - Elegido debido a su estatus predilecto como entorno integrado de desarrollo destacado para el lenguaje PHP. Dado que pertenece al mismo desarrollador que WebStorm, coincide en muchos aspectos de look & feel con éste, lo que redunda en una facilidad de aprendizaje y suma en productividad al equipo de desarrollo.
- **HeidiSQL 10** como gestor y administrador del motor de bases de datos relacionales.
 - Debido a su facilidad de conexión con prácticamente cualquier motor de base de datos y la sencilla y cómoda interfaz de usuario que presenta.

Pruebas

Para cada evento, se realizaron diversas pruebas del prototipo con los organizadores del mismo. Estas consistieron en las pruebas unitarias, de integración y de aceptación. En cada uno de los casos, los resultados finales fueron positivos.

Sin embargo, la prueba real del prototipo fue la utilización de la aplicación en el evento mismo. Esto permitió obtener información valiosa, a partir de la cual se realizaron ajustes y correcciones en los siguientes incrementos. Lo dicho supuso una ventaja enorme para el resultado final del sistema.

Tipos de pruebas

Pruebas unitarias

El término prueba unitaria, unit testing en el inglés original, se utiliza para definir a una forma estructurada de comprobar el correcto funcionamiento de una unidad de código¹⁰.

Convencionalmente las pruebas unitarias son de bajo nivel, concentrándose en partes pequeñas de un sistema de software. Estas pruebas son escritas por los desarrolladores utilizando sus herramientas regulares, con la salvedad de que se emplea algún tipo de *framework*. En nuestro caso, se optó por utilizar Jasmine.

Por otra parte, para poder evaluar cada una de las pruebas de software escritas, se utiliza una suite de testing. En esta oportunidad, se decidió emplear Karma.

Dentro del desarrollo del presente incremento que hace al proyecto de software aquí informado, hemos utilizado también casos de pruebas manuales. Esto permitió probar

¹⁰**Wikipedia, La enciclopedia libre (2020). Prueba unitaria.** Recuperado el 28 de noviembre de 2020, de https://es.wikipedia.org/wiki/Prueba_unitaria.

comportamientos de componentes visuales o ciertos requerimientos que no se pueden testear del todo a través de pruebas automáticas.

Pruebas Manuales

Un testing manual o prueba manual es un tipo de prueba de software donde los probadores ejecutan manualmente los casos de prueba sin usar ninguna herramienta de automatización¹¹.

Se incluye en el Anexo I un ejemplo de corrida de la suite de pruebas manuales.

Pruebas Automáticas

Convencionalmente las pruebas unitarias son de bajo nivel, concentrándose en partes pequeñas de un sistema de software. Estas pruebas son escritas por los desarrolladores utilizando sus herramientas regulares, con la salvedad de utilizar para el desarrollo de las pruebas algún tipo de *framework* para pruebas, ocupando este lugar Jasmine dentro de nuestro *stack* de tecnologías. Para poder evaluar cada una de las pruebas de software escritas, se utiliza una suite de testing. En nuestro caso, la herramienta utilizada a tal fin es Karma.

Prueba Correcta

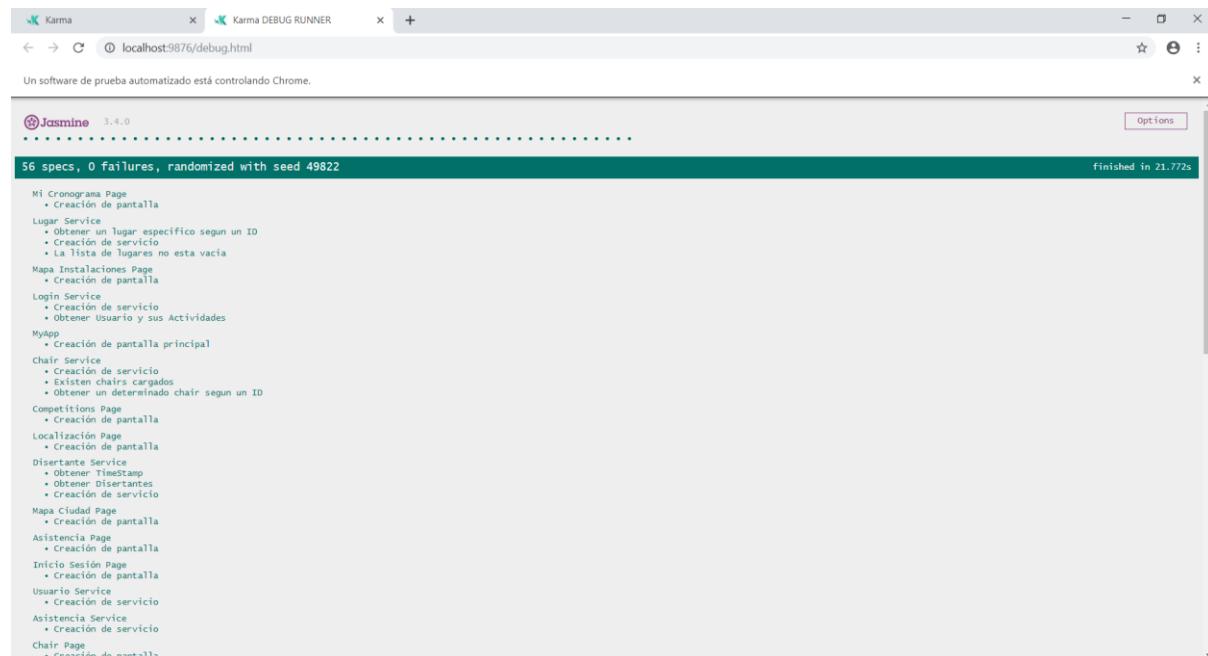


Figura 4 – Resultados correctos del test automático parte 1.

¹¹Cómo hacer testing de Software Manual. Recuperado el 22 de diciembre de 2020, de <https://www.tecnologias-informacion.com/testingmanual.html>.

A screenshot of a web browser window titled "Karma DEBUG RUNNER" displaying the URL "localhost:9876/debug.html". The page content is a list of test cases, each with a title and a list of assertions. The tests are categorized by service and page type. The assertions are mostly related to the creation and retrieval of objects like organizers, areas temáticas, and activities.

```
Un software de prueba automatizado está controlando Chrome.

Iricio Page
  • Creación de pantalla
Actividad Page
  • Creación de pantalla
Tabs Controller Page
  • Creación de pantalla
Competition Service
  • Creación de servicio
  • Obtener datos de competición
Organizador Service
  • La lista de organizadores no esté vacía
  • Obtener un organizador específico según un ID
  • Creación de servicio
Área Temática Page
  • Creación de pantalla
Diseñantes Page
  • Creación de pantalla
Diseñante Page
  • Creación de pantalla
Área Temática Service
  • Obtener una determinada área temática según un ID
  • Existen áreas temáticas cargadas
  • Creación de servicio
Marcador Service
  • Creación de servicio
  • La lista de marcadores no esté vacía
  • La lista de tipos de marcadores no esté vacía
  • Obtener un marcador específico según un ID
  • Obtener un tipo de marcador específico según un ID
Sponsors Page
  • Creación de pantalla
About Page
  • Creación de pantalla
Settings Page
  • Creación de pantalla
Sponsor Service
  • Obtener timestamp
  • Creación de servicio
  • Obtener sponsors
Próxima Actividad Page
  • Creación de pantalla
Actividad Service
  • Obtener tipos de actividad específico según un ID
  • Obtener timestamp
  • Creación de servicio
  • Obtener tipos de actividades
Perfil Page
  • Creación de pantalla
Notificaciones Service
  • Creación de servicio
Áreas Temáticas Page
  • Creación de pantalla
```

Figura 5 – Resultados correctos del test automático parte 2.

A screenshot of a web browser window titled "Karma DEBUG RUNNER" displaying the URL "localhost:9876/debug.html". The page content is a list of test cases, similar to Figure 5, but with some differences in the test titles and assertions. The tests are categorized by service and page type, with assertions related to the creation and retrieval of various entities.

```
Un software de prueba automatizado está controlando Chrome.

Organizador Service
  • La lista de organizadores no esté vacía
  • Obtener un organizador específico según un ID
  • Creación de servicio
Área Temática Page
  • Creación de pantalla
Diseñantes Page
  • Creación de pantalla
Diseñante Page
  • Creación de pantalla
Área Temática Service
  • Obtener una determinada área temática según un ID
  • Existen áreas temáticas cargadas
  • Creación de servicio
Marcador Service
  • Creación de servicio
  • La lista de marcadores no esté vacía
  • La lista de tipos de marcadores no esté vacía
  • Obtener un marcador específico según un ID
  • Obtener un tipo de marcador específico según un ID
Sponsors Page
  • Creación de pantalla
About Page
  • Creación de pantalla
Settings Page
  • Creación de pantalla
Sponsor Service
  • Obtener timestamp
  • Creación de servicio
  • Obtener sponsors
Próxima Actividad Page
  • Creación de pantalla
Actividad Service
  • Obtener tipos de actividad específico según un ID
  • Obtener timestamp
  • Creación de servicio
  • Obtener tipos de actividades
Perfil Page
  • Creación de pantalla
Notificaciones Service
  • Creación de servicio
Áreas Temáticas Page
  • Creación de pantalla
```

Figura 6 – Resultados correctos del test automático parte 3.

Prueba Fallida

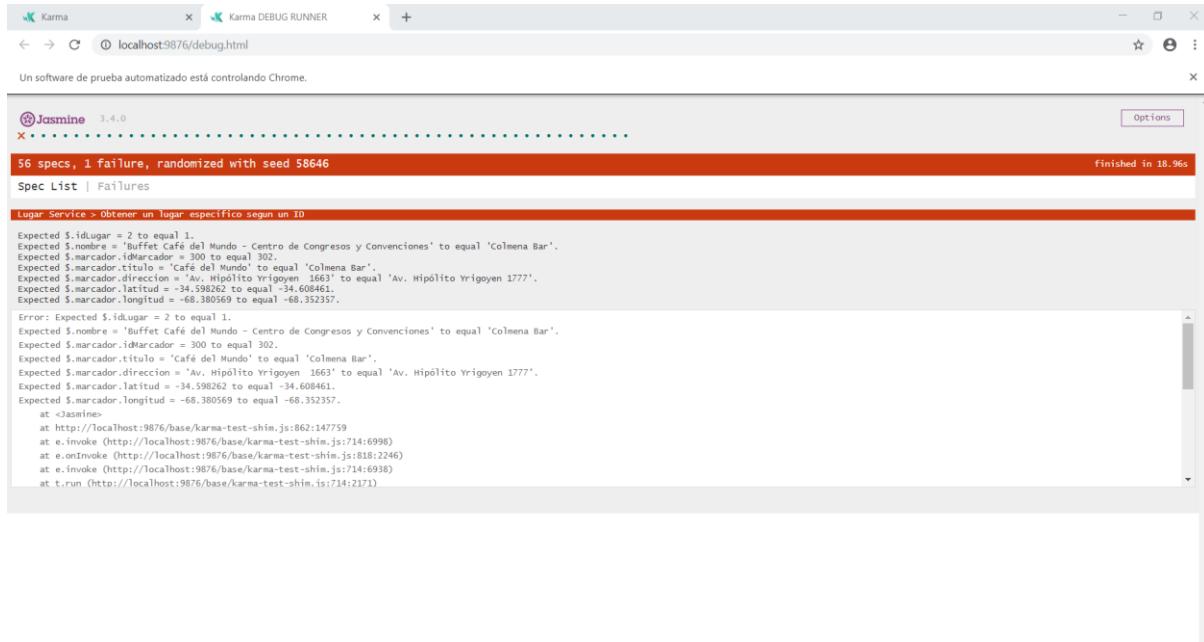


Figura 7 – Resultados fallidos del test automático.

Pruebas de aceptación

Las pruebas de aceptación son pruebas manuales, diseñadas por los desarrolladores y llevadas a cabo en conjunto con los clientes.

Estas consisten en el seguimiento de determinados pasos para comprobar el correcto funcionamiento de requerimientos, tanto funcionales como no funcionales dentro de las pantallas de la aplicación.

Herramientas

- Karma
 - Karma es un Test Runner, un entorno que permite ejecutar los test de Javascript según se vayan construyendo, de tal forma que ante cualquier fallo el desarrollador se dará cuenta de inmediato¹².
- Jasmine
 - Jasmine es un framework Javascript que es utilizado para la definición de tests usando un lenguaje natural entendible por todo tipo de personas¹³.

¹²Álvarez, C. (2015). **Karma.js, el concepto de Test Runner**. Recuperado el 26 de mayo de 2020, de <https://www.genbeta.com/desarrollo/karma-js-el-concepto-de-test-runner>.

¹³Coding Potions (2018). **Angular - Cómo hacer testing unitario con Jasmine**. Recuperado el 26 de mayo de 2020, de <https://codingpotions.com/angular-testing>.

Gestión del Proyecto

Herramientas

- **Git**, como control de versionado de código. Es un software de control de versiones destacable comúnmente usado en la industria del desarrollo de software. Se eligió, además de por sus características principales mencionadas, por la cantidad de clientes existentes que facilitan su uso y por la experiencia previa del equipo de desarrollo. Además, su elección se basó en la gran cantidad de clientes que utilizan este software. Por último, el equipo de desarrollo cuenta con experiencia previa satisfactoria ante la implementación de este.
- **Cliente de Git - Webstorm**, como cliente de git integrado en el IDE de desarrollo. Fue elegido debido a que permite ver los cambios que están en *stage*, agregar archivos excluidos y visualizar las ramas de los proyectos de forma fácil e intuitiva.
- **Lucidchart**, como herramienta de creación de diagramas. Es una herramienta web que permite crear diagramas de todo tipo, ofrece muchas plantillas y permite la colaboración en tiempo real. Se seleccionó debido a la experiencia previa satisfactoria de uno de los desarrolladores y por la versatilidad y facilidad de uso de esta.
- **Github**, como plataforma de desarrollo colaborativo, la cual utiliza git como herramienta de versionado. Se seleccionó debido a la experiencia previa de los desarrolladores utilizando la plataforma.
- **Trello**, como tablero Kanban virtual. Es una aplicación de gestión de proyectos que proporciona un “tablero virtual” que se puede utilizar para organizar tanto proyectos personales como en equipo. Su elección se fundamentó en la facilidad de uso y en la experiencia previa del equipo con la herramienta.
- **Google G Suite**, como herramienta de almacenamiento de información externa al código y al software en general. Es un servicio de almacenamiento en la nube que admite cualquier tipo de archivos e incluso permite generar otros archivos a través de las herramientas de ofimática que proporciona. Se seleccionó debido a que el equipo de desarrollo ya contaba con una cuenta registrada en esta plataforma.
- **Discord**, como herramienta de comunicación del equipo de desarrollo. Es una plataforma de comunicación tanto por texto o voz, donde cada usuario puede contactarse con otro por privado o crear servidores donde hay múltiples usuarios comunicándose al mismo tiempo. Se eligió debido a la excelente calidad que proporciona el servicio.
- **Skype**, como herramienta de comunicación con los clientes. Es una herramienta de comunicación que permite la comunicación por texto, voz y/o video. Fue utilizada debido a que los clientes previamente la tenían instalada y sabían cómo usarla.

Desarrollo del sistema

En esta sección se detallan los módulos que proveen la funcionalidad esperada en el sistema, incluyendo en su descripción si estas funcionalidades están disponibles para organizadores, para usuarios, o para ambos roles. El sistema en cuestión arquitectónicamente se separa en una aplicación móvil cliente y en un servidor web, con las características que provee la aplicación móvil trabajando en consonancia con los servicios expuestos por el servidor web para proveer las funcionalidades del software.

Debido a que el desarrollo se planteó de forma incremental, se utilizó un diseño modular para la aplicación. Esto quiere decir que dentro del desarrollo del sistema un módulo puede evolucionar o habilitarse independientemente del resto, siendo la disponibilidad de cada módulo parte de la configuración global del sistema. Cada módulo puede contener una o más pantallas y, a su vez, puede estar relacionado con otros módulos enlazando las pantallas de estos por diferentes medios (los más comunes son botones).

A continuación. se desarrollarán brevemente los módulos incluidos en la aplicación correspondiente a este PFC.

Listado de módulos desarrollados

Inicio de Sesión

Módulo funcional para los asistentes y organizadores. Para poder iniciar sesión en la aplicación móvil es necesario estar previamente registrado. No obstante, existe también la posibilidad de acceder como invitado, es decir, sin disponer de una cuenta, aunque con ciertas limitaciones, como ser el acceso a la asistencia, datos del perfil y cronograma personalizado.

Mi Cronograma

Consiste en un listado de las actividades con distintos tipos de filtros y un detalle de estas cuando se las presiona.

Próxima Actividad

Permite ver la actividad actual y la próxima a desarrollarse, pudiendo acceder a un detalle de la misma en el que se visualizan datos como el tipo de la actividad, nombre, descripción, hora de inicio y de fin, moderadores, disertantes, la dirección, un mapa y documentos de los trabajos relacionados con la actividad.

Actividades Favoritas

Sirve para definir cuáles son las actividades favoritas por parte del asistente en el módulo de Mi Cronograma. Se muestra una lista de las seleccionadas.

Asistencia

Utilizada para 1) la toma de asistencia por actividad a través de un código QR generado por la aplicación móvil y 2) la persistencia de las asistencias registradas en el servidor web. A partir del registro de asistencias luego los organizadores pueden métricas y estadísticas de lo acontecido durante el evento.

Disertantes

Lista los disertantes e información sobre ellos, así como también una lista de las actividades en las que participa cada uno.

Localización

Contiene el plano de las instalaciones en donde es llevada a cabo la conferencia, un mapa de la ciudad y lista de lugares de interés para el asistente (hoteles, restaurantes, paradas de autobús, lugares de entretenimiento, entre otros), y aplicaciones referidas a la movilidad.

Ajustes

Se trata de un espacio donde es posible configurar ajustes generales que afectan a toda la aplicación. Dentro de lo estipulado para los requerimientos del proyecto actual, este módulo debe soportar la elección del idioma en el cual se presentará la aplicación.

Sponsors

Presenta un carrusel con los logos de los sponsors en la pantalla de inicio, así como también los presenta en forma de lista en una pantalla específica. Además, se puede acceder a un detalle cuando se lo selecciona.

Perfil

Muestra los datos del asistente junto a su código QR personal para la asistencia y un botón para cerrar sesión.

Acerca de

Contiene información referida a los desarrolladores de la aplicación.

Ambiente de Desarrollo

Dentro del sistema, se presentan distintos ambientes de desarrollo. Esto significa que existen archivos, bases de datos y APIs diferenciadas e independientes que se usan para distintos propósitos. Esto resulta de especial importancia porque, por ejemplo, se pueden implementar nuevas funcionalidades en el sistema sin afectar a la aplicación funcionando, o se puede probar cómo se comportaría un cambio cierta funcionalidad de la aplicación antes de que llegue a afectar a los usuarios reales del sistema.

Por lo general existen tres ambientes de desarrollo: local, staging o pre producción y producción.

Local

El ambiente de desarrollo local es aquél en el que se realiza la completitud del desarrollo del sistema, y se prueba y depura el mismo.

Para este proyecto en específico, se ha utilizado el IDE mencionado anteriormente. A su vez, se empleó el comando “ionic serve” del framework para ejecutar localmente la aplicación en un navegador web, acompañado con la consola de Google Chrome para la depuración y pruebas.

Staging o Pre Producción

Consiste en un ambiente en el que existe una réplica exacta del ambiente de producción, sólo que los usuarios no están activos en este ambiente. Se utiliza para probar cómo se comportaría alguna modificación del sistema en producción, pero sin afectarlo.

Para este, se hacían deploys para el servidor web, para el cliente web, para Android haciendo uso de Android SDK y para iOS haciendo uso del emulador incluido en xCode.

Producción

El ambiente de producción es aquel que se utiliza para los usuarios del sistema, es “lo que ve el usuario”. Este ambiente es muy delicado debido a que cada cambio realizado afecta directamente a los usuarios del sistema, por eso es importante el uso de los ambientes de desarrollo anteriores.

Para este ambiente, se hubieran realizado deploys para el servidor web de producción, para el cliente web de producción, para Android haciendo uso de Google Play Console y para iOS haciendo uso de Apple AppStore Connect. En los dos últimos casos, se hubiera realizado un despliegue y posterior publicación de las aplicaciones móviles en estas herramientas que son específicas de las plataformas Android y iOS.

Prototipos previos

Como se menciona en la introducción del presente documento, existe una serie de prototipos previos realizados para eventos académicos reales que se realizaron en la ciudad de Santa Fe.

Gracias a la realización de los mismos, el equipo de desarrollo logró un aprendizaje sobre la tecnología, ahorrando mucho tiempo en lo que respectaría una curva de aprendizaje. Además, se logró definir previamente cuestiones como la arquitectura a utilizar, la preparación del proyecto en el IDE, los repositorios, servidores, entre otras cosas referidas a configuración del proyecto en general. También, los módulos de los que se parten fueron refactorizados durante cada uno de los prototipos.

A continuación, se mencionan las funcionalidades previstas para cada prototipo incremental.

Prototipo base

Prototipo CAEII 2018

- Inicio de sesión de los asistentes
- Tema personalizado (fondos y colores)
- Menús de navegación
- Visualización del cronograma completo
- Acceso a un detalle de cada actividad

Incrementos previos

JEA 7

- Inicio de sesión de los asistentes
- Tema personalizado (fondos y colores)
- Menús de navegación
- Visualización del cronograma completo
- Acceso a un detalle de cada actividad
- Visualización de disertantes de las actividades
- Acceso a un detalle de cada disertante
- Integración con Whatsapp para comunicación con organización
- Acceso a una encuesta sobre el evento

CONAIISI 2017

- Inicio de sesión de los asistentes
- Tema personalizado (fondos y colores)
- Menús de navegación
- Visualización del cronograma completo
- Acceso a un detalle de cada actividad
- Visualización de disertantes de las actividades
- Integración con Whatsapp para comunicación con organización

- Acceso a un detalle de cada disertante

Pre-CAEII 2018

- Inicio de sesión de los asistentes
- Tema personalizado (fondos y colores)
- Menús de navegación
- Visualización del cronograma completo
- Acceso a un detalle de cada actividad
- Visualización de disertantes de las actividades
- Acceso a un detalle de cada disertante

CAEII 2018

- Inicio de sesión de los asistentes
- Tema personalizado (fondos y colores)
- Menús de navegación
- Visualización del cronograma completo personalizado por asistente
- Acceso a un detalle de cada actividad
- Visualización de disertantes de las actividades
- Acceso a un detalle de cada disertante

ITU Kaleidoscope 2018 + World Smart Cities Forum

- Inicio de sesión de los asistentes
- Tema personalizado (fondos y colores)
- Menús de navegación
- Visualización del cronograma completo
- Acceso a un detalle de cada actividad
- Visualización de disertantes de las actividades
- Acceso a un detalle de cada disertante

Arquitectura

Debido a la naturaleza móvil de la aplicación que hace al presente proyecto, varios aspectos fueron los contemplados a la hora de formular la arquitectura que hace al mismo. Estos fueron:

- minimizar el tráfico de datos, a fin de reducir la necesidad de conectarse al servidor debido a la falta de conectividad en algunos lugares;
- minimizar el consumo de batería, para que los asistentes puedan usar la aplicación sin que afecte al uso normal del teléfono, debido a la escasez de acceso a sitios para cargar el móvil;
- reducir al mínimo el tamaño de la aplicación, para que los usuarios la puedan instalar sin la necesidad de eliminar de la memoria otras aplicaciones o archivos.

Una de las características de Ionic, es que no se puede desarrollar la parte del cliente y el servidor unidas en la misma aplicación, sino que se está obligado a implementar una arquitectura tipo Cliente/Servidor. Para la comunicación entre el cliente y el servidor, se optó por la utilización de servicios web RESTful, comunicando a ambos mediante el uso de una API.

Una API RESTful es una interfaz entre sistemas que usa HTTP para obtener datos o generar operaciones sobre los mismos en todos los formatos posibles¹⁴. Es una alternativa a otros protocolos estándar de intercambio de datos como SOAP (Simple Object Access Protocol).

En el Anexo III se puede encontrar el diagrama de arquitectura de la aplicación donde se puede ver la estructura mencionada.

Cliente

En el caso de este proyecto, el rol del cliente de la arquitectura lo toma la aplicación móvil, generando solicitudes a la API expuesta por el servidor web e interpretando las respuestas para ya sea mostrar información, guardar información o enviarla información al servidor.

Servidor

En este caso, el papel del servidor lo toma la API RESTful implementada en PHP y alojada en un servidor en la nube. Este toma las peticiones que hace el cliente, las interpreta, hace lo solicitado según los parámetros de entrada y devuelve una respuesta, que puede ser información o alguna confirmación de que se ejecutó correctamente.

¹⁴**BBVA API Market (2016). API REST: qué es y cuáles son sus ventajas en el desarrollo de proyectos.** Recuperado el 7 de junio de 2020, de <https://www.bbvaapimarket.com/es/mundo-api/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos/>.

Deployment

El despliegue del sistema es realizado a través de los comandos correspondientes de Ionic y Cordova, generando una versión para cada plataforma (Web, Android e iOS) y luego subido al servidor web o a la correspondiente tienda.

En el Anexo II, se pueden encontrar las instrucciones para realizar el deploy en cada plataforma.

Se considera importante destacar que para un próximo incremento sería interesante y más cómodo disponer de un ambiente de integración continua (CI/CD) y automatizar los deploys para el ambiente de staging y producción.

Diagrama de arquitectura

Representa la arquitectura del sistema de software

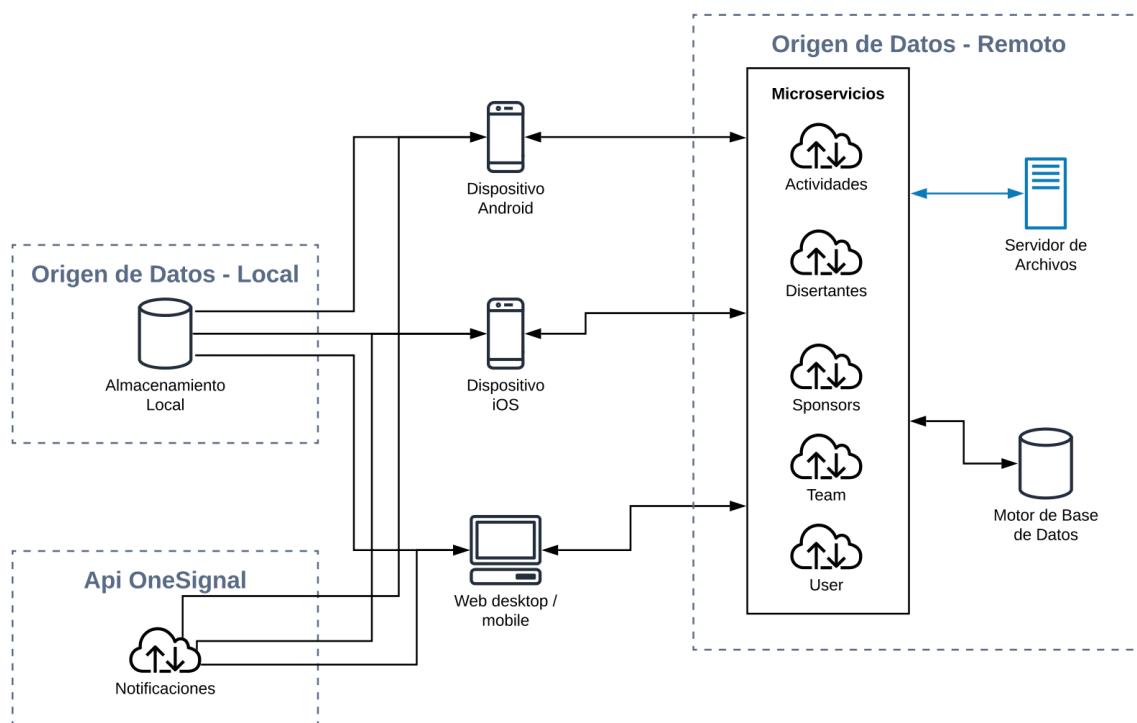


Figura 8 – Diagrama de arquitectura¹⁵.

¹⁵Se puede encontrar en formato digital en
<https://github.com/federalbertengo/Conferentia/blob/master/Informe/Modelos/Diagrama%20de%20Arquitectura.pdf>.

Requerimientos del incremento

Para el incremento realizado para este proyecto, han surgido muchos requerimientos tanto funcionales como no funcionales que se han implementado satisfactoriamente.

Requerimientos funcionales

Asistencia

A partir de un perfil de usuario y un código QR, debe analizarse, diseñarse e implementarse un mecanismo eficaz y eficiente para la toma de asistencia en cada una de las charlas y actividades que forman parte del congreso. Mediante el uso de este módulo, los usuarios organizadores son capaces de determinar, previamente, cuáles son las actividades dentro del evento que soportan y/o requieren la toma de asistencia, según un criterio por ellos establecido. De esta manera, la aplicación móvil permitirá la toma de asistencia para actividades determinadas.

Geolocalización

La aplicación debe proveer un módulo que contenga un mapa de la ciudad, con la señalización de las ubicaciones geográficas de importancia para quienes asistan al congreso, incluyendo para ello diferentes puntos de vista del mapa.

Se deben incorporar links para la descarga de las aplicaciones “Cuándo Pasa” y “Cómo Voy”, para informarse acerca de las opciones de transporte público.

Perfil de usuario

Cada usuario debe poseer su perfil propio con algunos datos personales de interés, a definir por los organizadores del evento.

Debe poder visualizar su respectivo porcentaje de asistencia a las actividades.

Debe incluirse la posibilidad de acceso a las características de la aplicación con un usuario “invitado”, el cual no puede validar su asistencia a las actividades del evento ni puede tener actividades específicas asignadas.

Auspiciantes

La aplicación debe proveer la posibilidad de incluir en la pantalla principal los auspiciantes del evento, de manera de monetizar el desarrollo de la aplicación mediante el ofrecimiento del espacio publicitario digital. Se debe incluir una lista de todos los auspiciantes del evento.

Notificaciones

Es necesario brindar la capacidad de recibir notificaciones de índole general, las cuales serán utilizadas para informar a los asistentes acerca de novedades, cambios de cronograma o informaciones relevantes durante el desarrollo del evento.

Próxima actividad

Se debe poder visualizar en un menú cuál es la próxima actividad dentro del cronograma que el usuario tiene asignada.

Actividades favoritas

La aplicación debe brindarle al usuario la posibilidad de fijar cuáles son las actividades de las que desea estar pendiente. Se debe proveer una visualización de la lista de las actividades favoritas seleccionadas.

Chairs, Moderadores y Organizadores

Es necesario poder visualizar, con foto y un breve CV, información acerca de las personas que ofician como chairs de sesión, moderadores de charlas y organizadores del evento.

Información de trabajos presentados

Se debe brindar acceso a los informes de los trabajos de investigación presentados en el evento. Debe poder accederse a los archivos de informes en formato PDF desde la vista detallada de actividad dentro del cronograma.

Requerimientos no funcionales

Capacidad de modificación de datos de las actividades

Esto ha sido resuelto a través de la implementación de archivos JSON en el servidor con la información de las actividades y un timestamp de la última modificación. De esta forma, cuando el usuario entra a la aplicación, se verifica el timestamp, y de ser necesario se actualiza la información.

Fluidez en el cronograma

Para esto se utilizó un componente llamado “virtualScroll”, que carga los datos en la lista a medida que la misma es desplazada.

Modelado

En esta sección se detallan cuáles fueron los distintos modelos, estructurales y conceptuales, que el equipo utilizó para documentar la lógica de negocios del sistema, explicando la interrelación entre las diversas funcionalidades y módulos del sistema.

Modelo de clases

Para la aplicación se ha utilizado un modelo relacional y de clases.

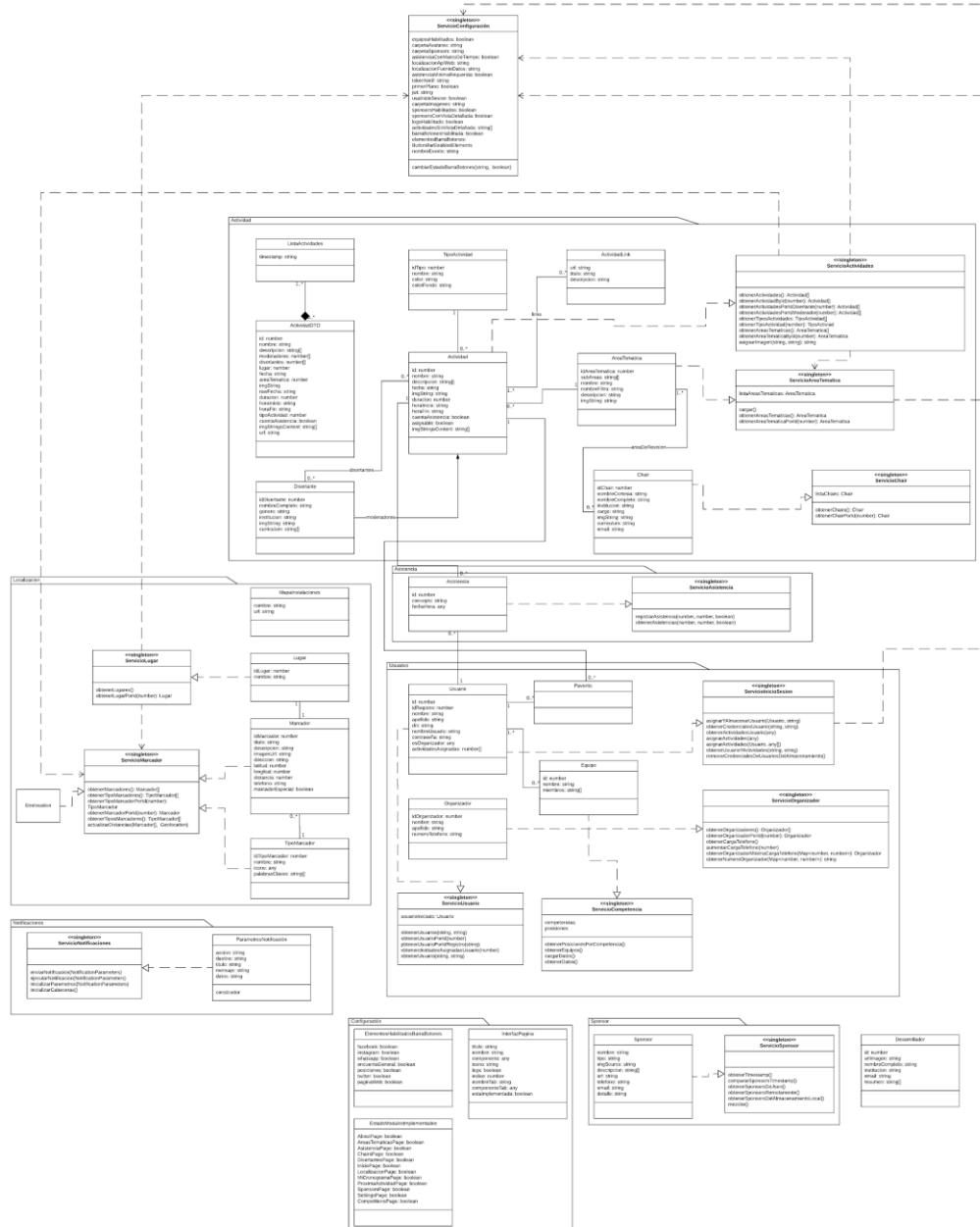


Figura 9 – Diagrama de clases empaquetado completo¹⁶.

¹⁶Se puede encontrar en formato digital en

Se puede encontrar en formato digital en:
<https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Modelos/Diagrama%20de%20Clases.pdf>.

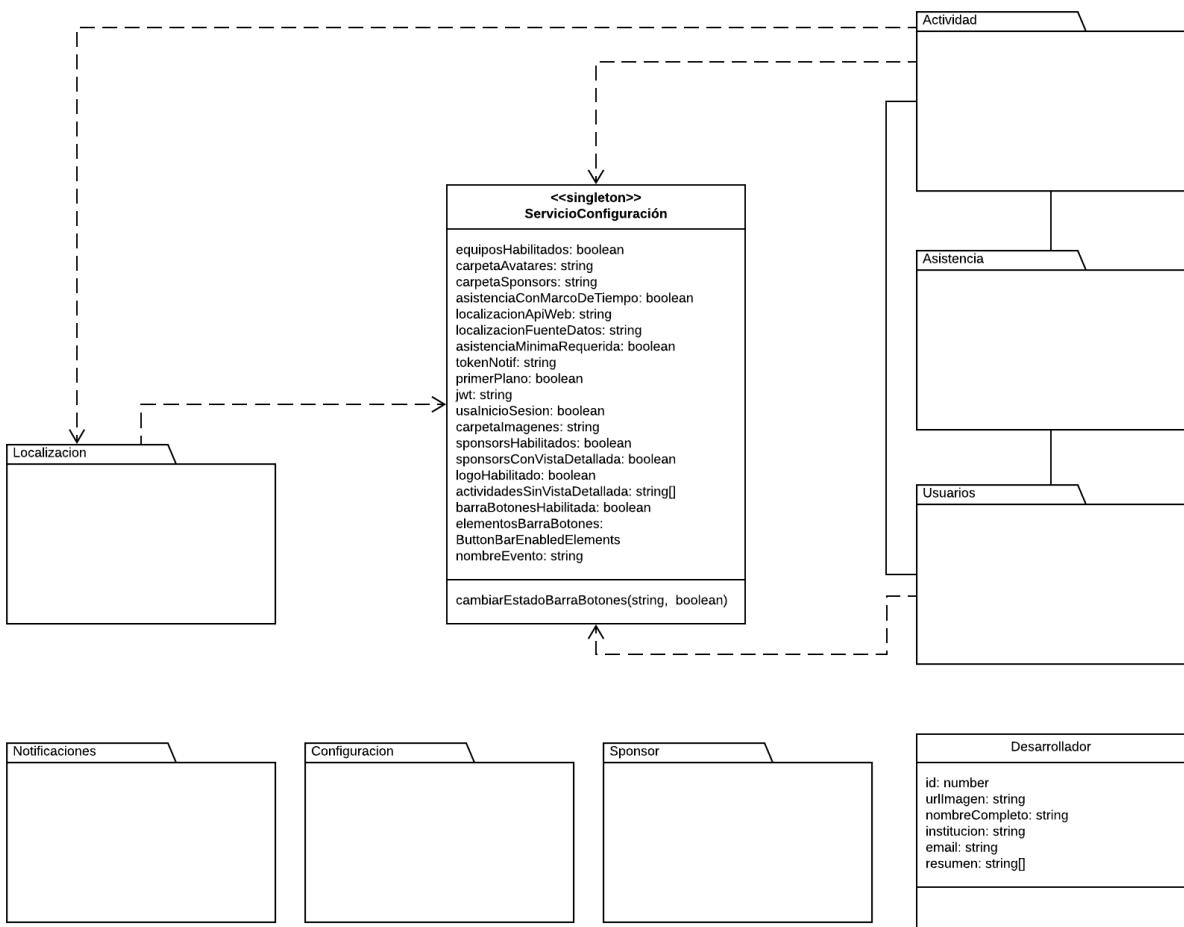


Figura 10 – Diagrama de clases empaquetado reducido¹⁷.

¹⁷Se puede encontrar en formato digital en
<https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Modelos/Diagrama%20de%20Clases.pdf>.

Diagramas de flujo

Estos modelos pueden explicar los pasos existentes en un proceso y su flujo desde el comienzo hasta el final.

Toma de Asistencia

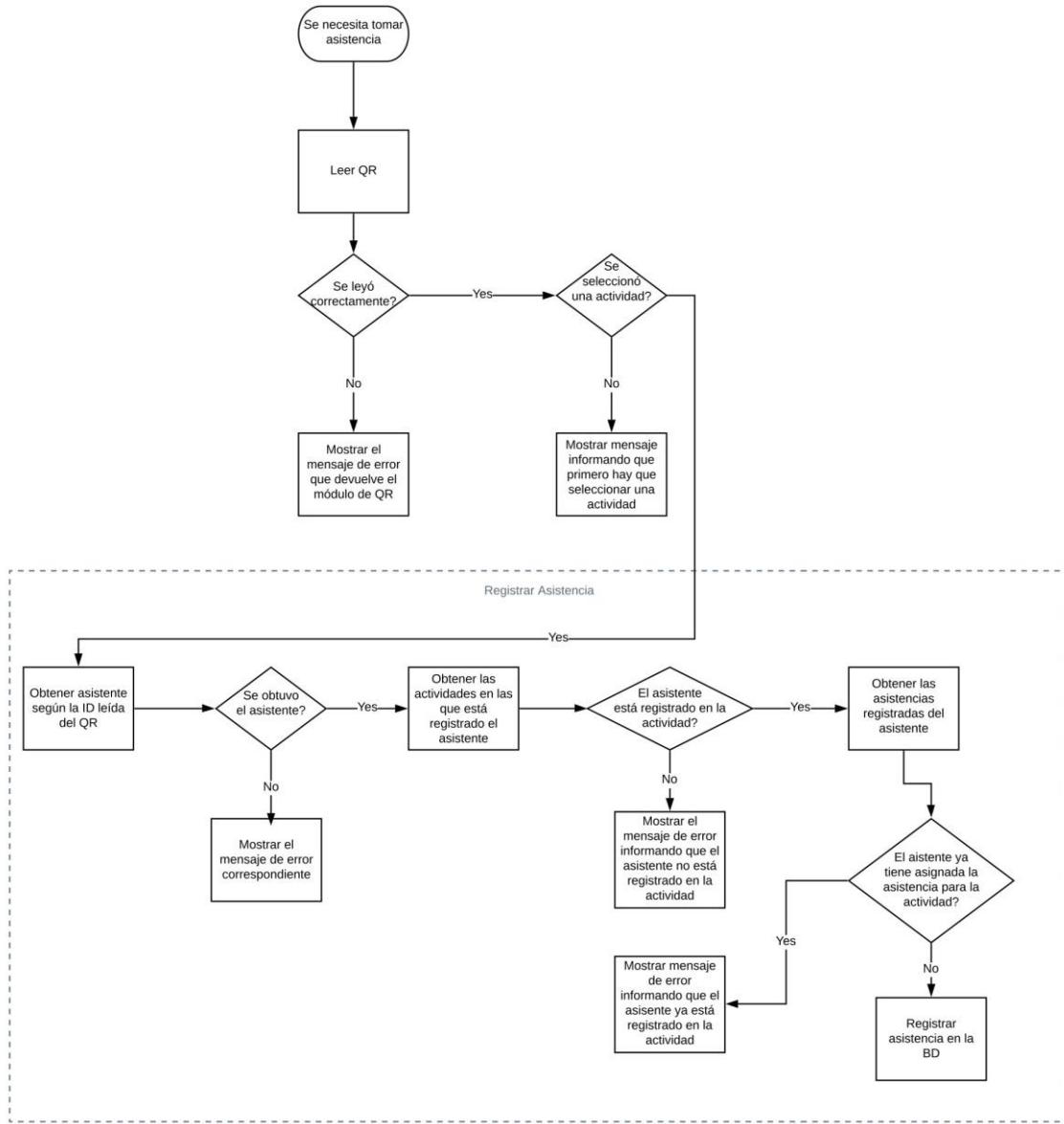


Figura 11 – Diagrama de flujo sobre el proceso de toma de asistencia¹⁸.

¹⁸Se puede encontrar en formato digital en <https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Modelos/Diagramas%20de%20Flujo.pdf>.

Balance de carga telefónica

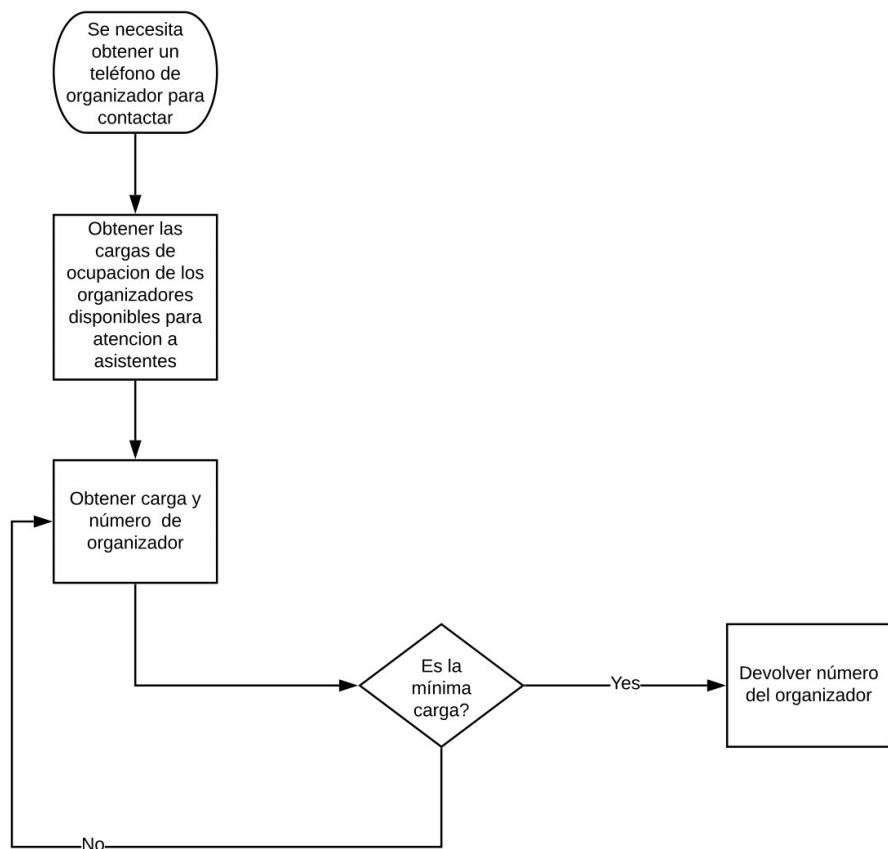


Figura 12 – Diagrama de flujo sobre el proceso de balanceo de carga telefónica¹⁹.

¹⁹Se puede encontrar en formato digital en <https://github.com/federalbertengo/Conferentia/blob/master/Informe/Modelos/Diagramas%20de%20Flujo.pdf>.

Obtención de datos locales y remotos

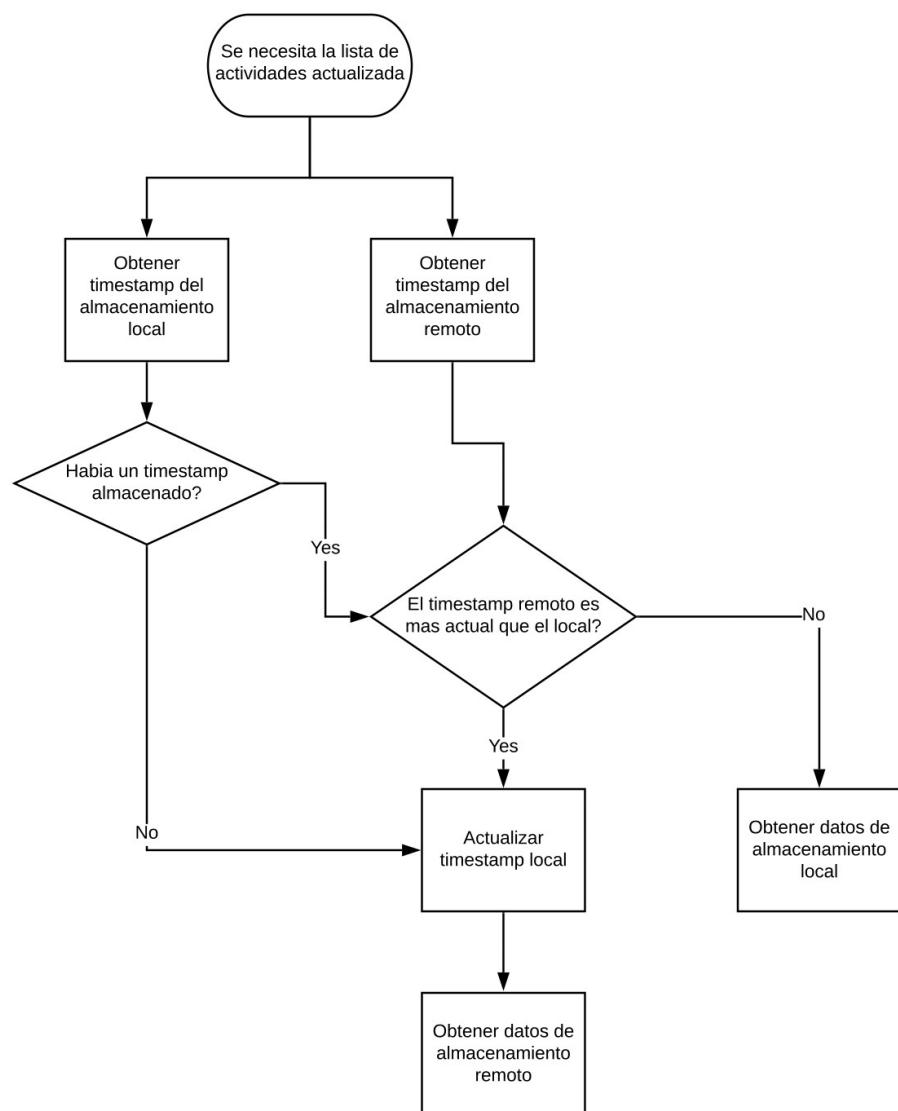


Figura 13 – Diagrama de flujo sobre el proceso de obtención de datos locales y remotos²⁰.

Diagramas de secuencia

Estos modelos sirven para explicar la interacción entre los componentes de un proceso determinado.

²⁰Se puede encontrar en formato digital en
<https://github.com/federalbertengo/Conferentia/blob/master/Informe/Modelos/Diagramas%20de%20Flujo.pdf>.

Obtención de datos locales y remotos

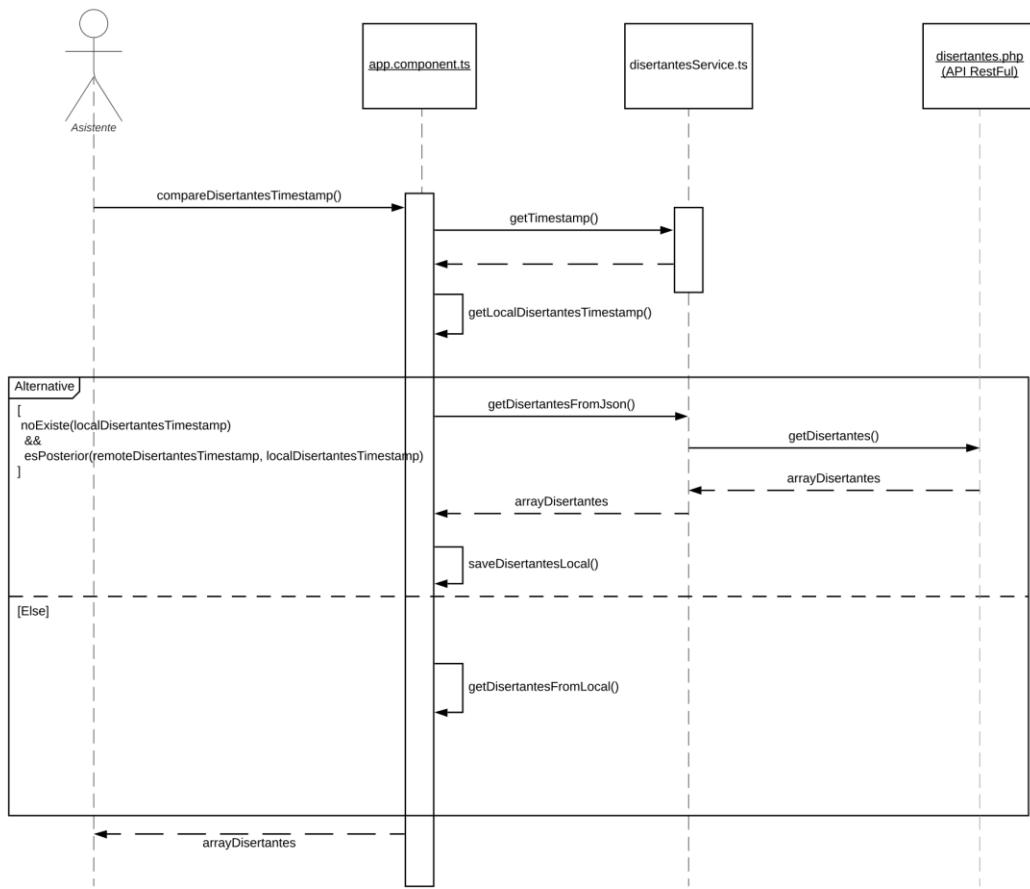


Figura 14 – Diagrama de secuencia sobre el proceso de obtención de datos locales y remotos²¹.

²¹Se puede encontrar en formato digital en <https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Modelos/Diagramas%20de%20Sencuencia.pdf>.

Toma de asistencia

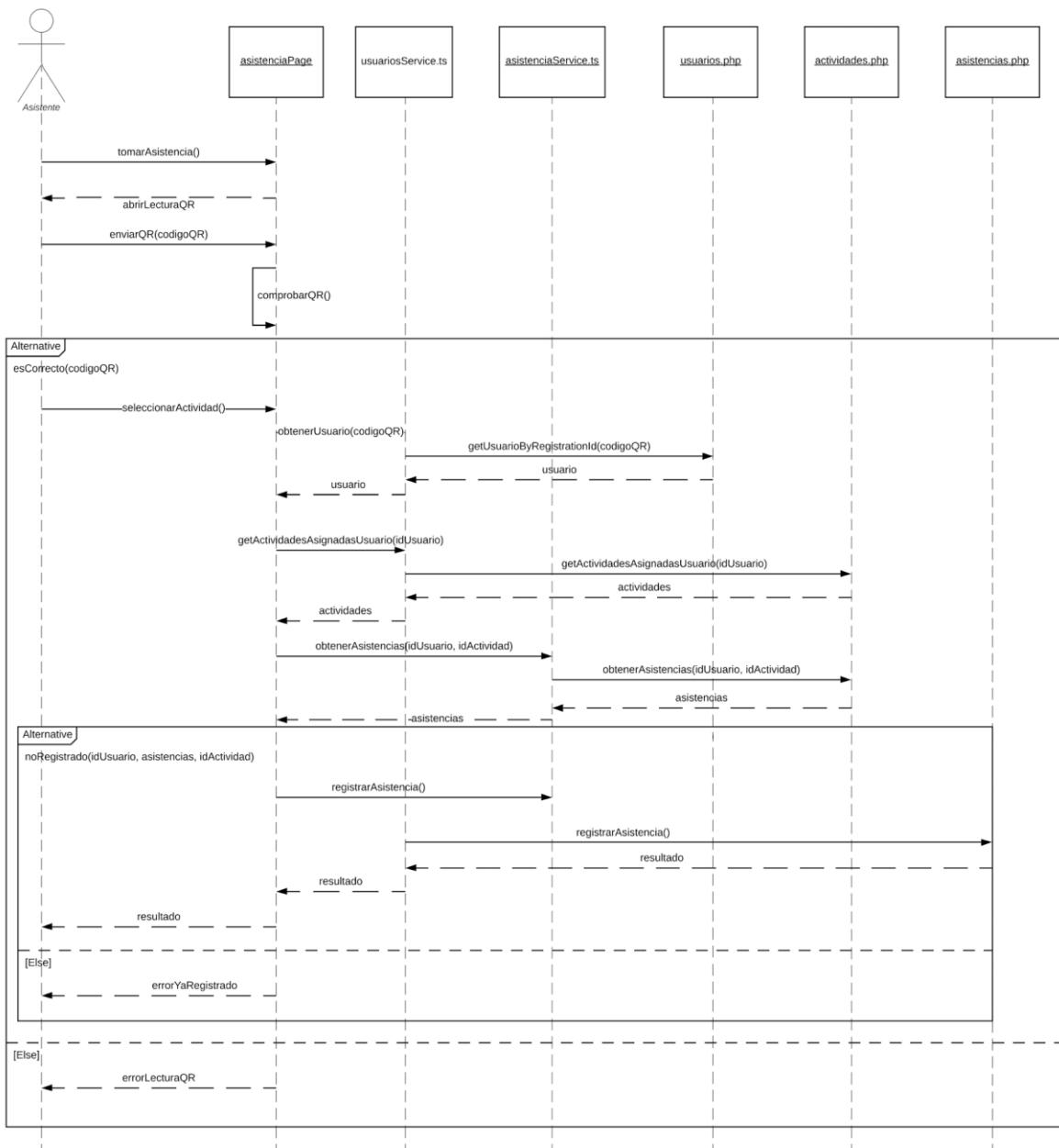


Figura 15 – Diagrama de secuencia sobre el proceso de toma de asistencia²².

Diagrama de despliegue

Este diagrama es utilizado para representar la distribución física de los componentes de software de un sistema.

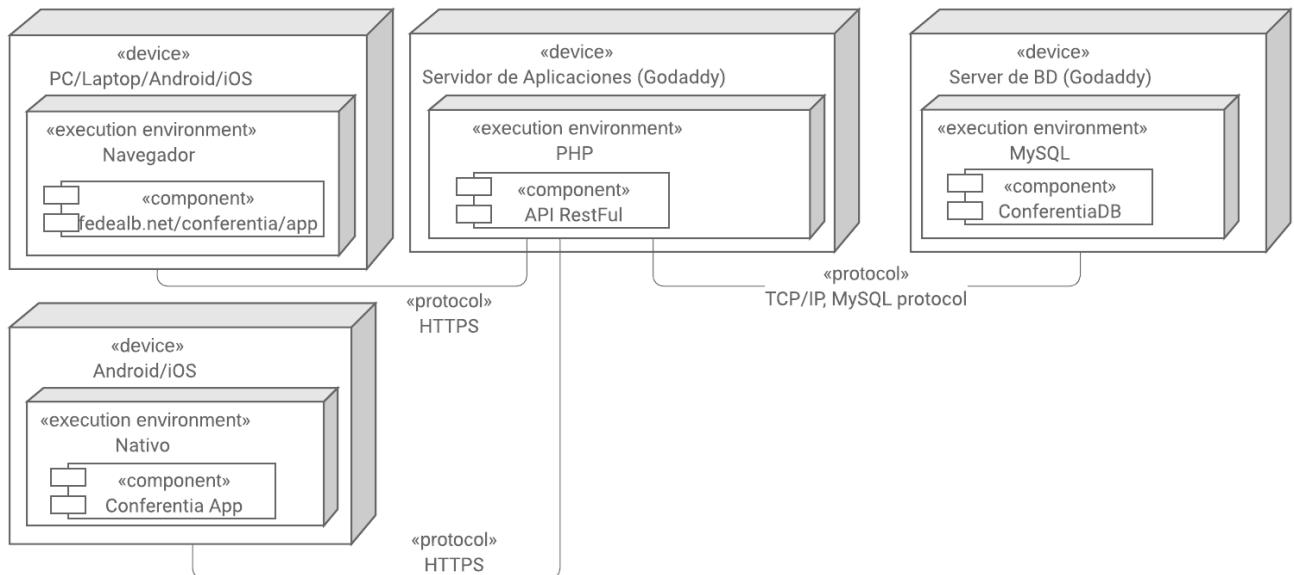


Figura 16 – Diagrama de despliegue²³.

Diagrama de entidad-relación

Representa la estructura de la base de datos y sus componentes. Una cuestión que vale la pena destacar sobre este diagrama, son los campos de auditoría, que no están representados en el mismo. Los mismos sirven para almacenar ciertos datos como la fecha de creación, fecha de modificación, si está eliminado y si está habilitado con propósito de mantener un control del estado y la validez de los registros de las tablas.

²²Se puede encontrar en formato digital en
<https://github.com/fdealbertengo/Conferentia/blob/master/Informe/Modelos/Diagramas%20de%20Secuencia.pdf>.

²³Se puede encontrar en formato digital en
<https://github.com/fdealbertengo/Conferentia/blob/master/Informe/Modelos/Diagrama%20de%20despliegue.pdf>.

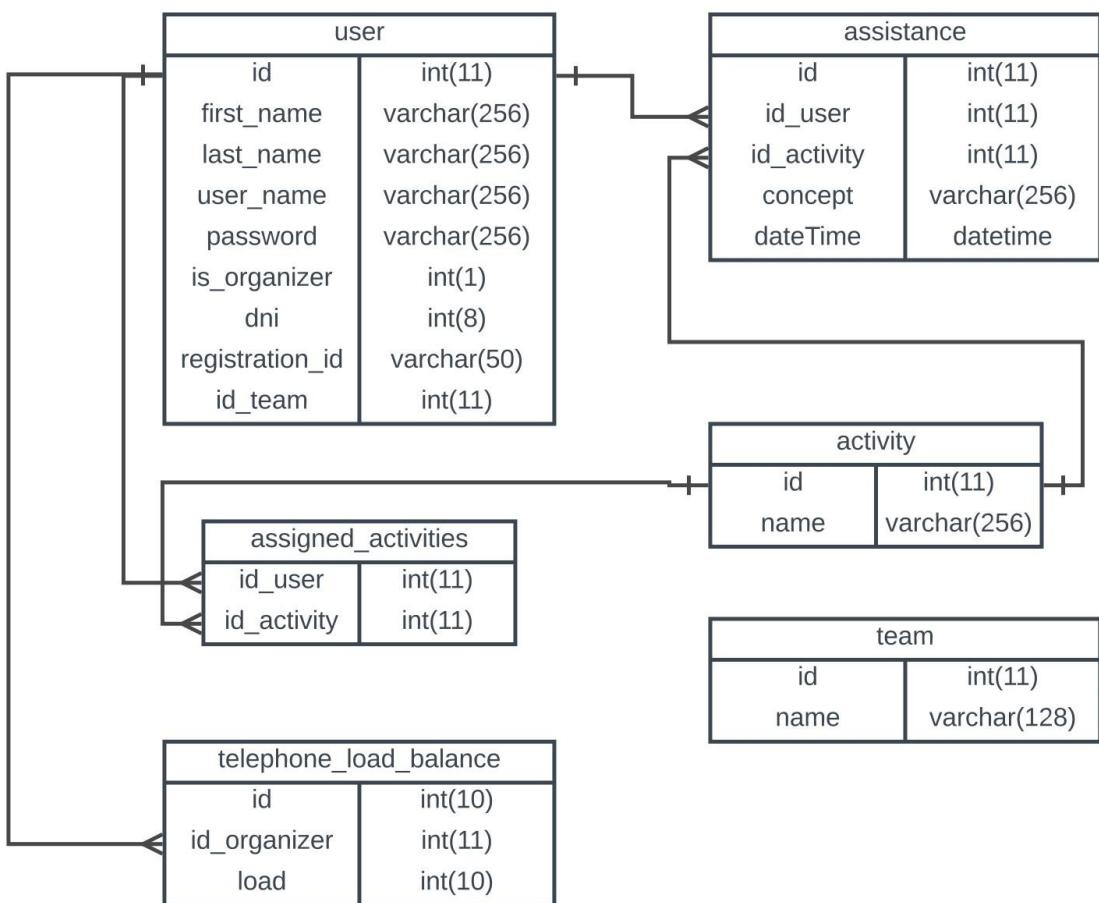


Figura 17 – Diagrama de entidad-relación del prototipo²⁴.

²⁴Se puede encontrar en formato digital en
<https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Modelos/Diagrama%20de%20entidad-relacion.pdf>.

Incremento

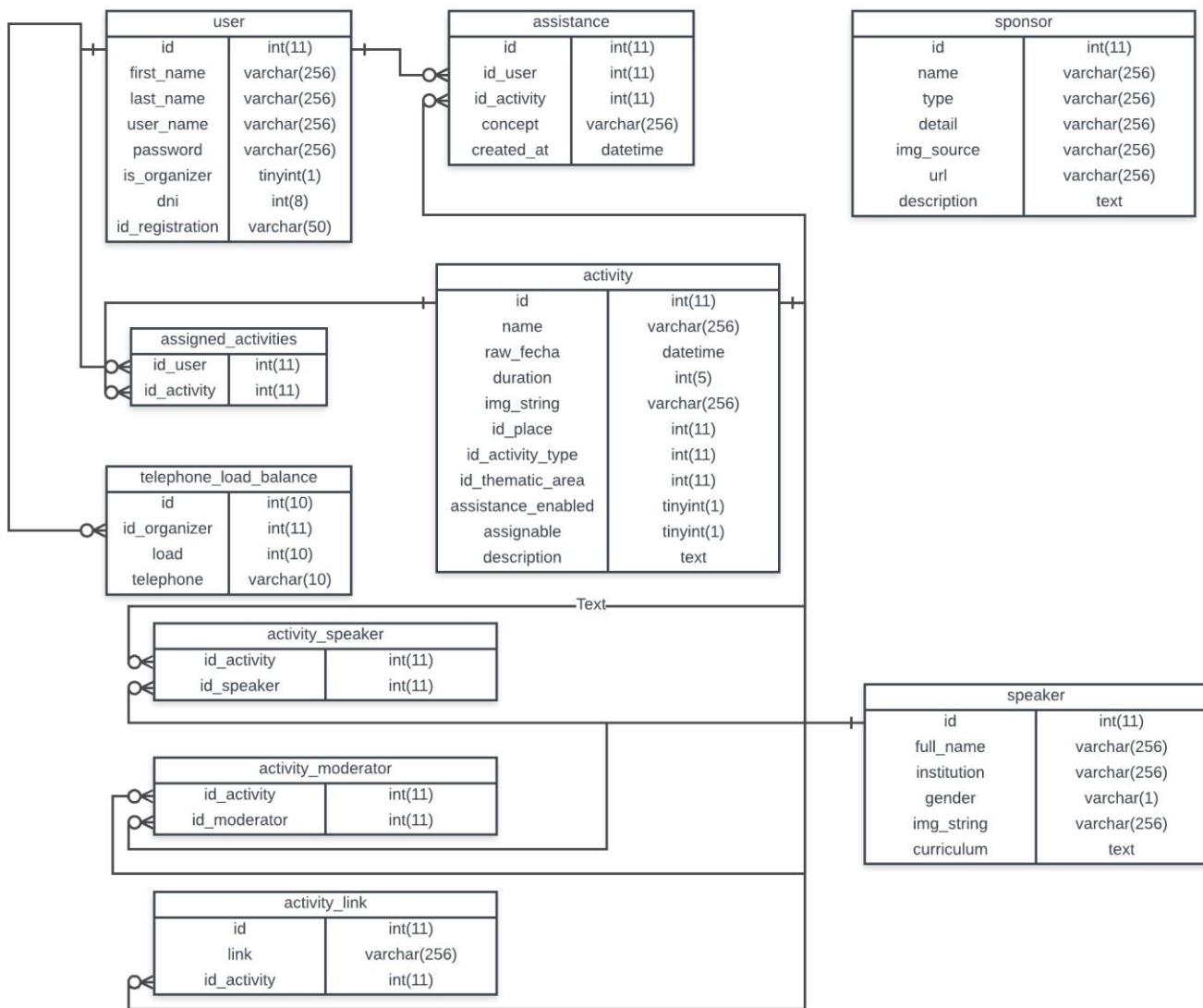


Figura 18 – Diagrama de entidad-relación en el incremento final²⁵.

Actividades y tareas

Como se mencionó anteriormente, se utilizó una metodología basada en Kanban para la gestión de tareas. La parte teórica de este método ya ha sido explicada en la sección [Proceso de Desarrollo](#). A continuación se detalla el procedimiento de su aplicación y cuál fue la experiencia utilizandoarlo.

En las reuniones con el representante de los stakeholders, surgían requerimientos tanto funcionales como no funcionales. Estos eran divididos en tareas que se agregaban a la

²⁵Se puede encontrar en formato digital en <https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Modelos/Diagrama%20de%20entidad-relacion.pdf>.

correspondiente columna del tablero. Luego, eran asignados a cada miembro del equipo de desarrollo según su experiencia y capacidades particulares. Esas tareas eran tomadas y realizadas según una prioridad que se determinaba al agregarla a la lista.

Por último, al finalizar una tarea, esta era movida a una columna determinada. Dichas columnas eran organizadas por incremento, semana o mes, según se consideraba pertinente. Finalmente, cuando la columna ya no era más requerida, se archivaba.

Gestión del Proyecto

Durante el desarrollo de una iniciativa²⁶ de software²⁶ resulta útil contar con métricas de software que permitan conocer la *salud* de la misma a lo largo de las diferentes fases del desarrollo. Para mantener un proceso de desarrollo ordenado que permita conocer estados presentes y anteriores en lo que a salud de la iniciativa se refiere, el poseer información actualizada y relevante sobre métricas de software también se transforma en una condición necesaria.

En esta sección se procede a describir cuáles fueron las métricas de software analizadas para evaluar el desempeño del equipo a lo largo del desarrollo de la iniciativa que nuclea al presente proyecto. Para la evaluación de la gestión del proyecto a lo largo del proceso de desarrollo el equipo hizo uso de los indicadores *alpha* definidos en el framework Essence, el cual hace uso del SEMAT Kernel. Para este proyecto, específicamente, se utiliza la versión 0.3 del mencionado framework.

Como parte de la presente sección, se procede primero a introducir al lector al framework Essence, el cual puede resultar una novedad dentro del campo de la Ingeniería de Software, por ser de especificación y definición relativamente reciente.

Posteriormente se procede a realizar una evaluación de lo planificado, especificando y explicando cuál es el estado de la iniciativa al comienzo del desarrollo y mostrando el progreso, según las métricas prescriptas por el framework, para las iteraciones definidas dentro de la planificación del PFC, conjuntamente con el despliegue de los entregables que forman parte de cada una de las iteraciones. De esta manera, se logra realizar un análisis comparativo respecto de lo que primeramente el equipo planificó y ejecutó, exponiendo las diversas dificultades y desafíos no contemplados en el plan de proyecto, y que más tarde se vieron reflejadas a la hora de realizar la ejecución del proyecto propiamente dicha.

²⁶ Jacobson et al. (2015) introduce el término *software endeavor*, traducible a “iniciativa de software” como término paraguas para englobar aquellas acciones o conjuntos de acciones que se toman con el propósito de cumplir un objetivo que involucre la generación de valor, acorde a las oportunidades existentes, para satisfacer los requerimientos de uno o más *stakeholders* mediante un sistema de software.

Framework Essence

Dentro del framework Essence, se introduce el término salud para una iniciativa de software (*health of a software endeavor*, por sus siglas en inglés) como una métrica multidimensional que permite, de forma cuantificable, asignar valores a atributos clave. Estos valores clave, en Essence, se denominan *alphas*.

Los *alphas* tienen la finalidad de hacer visible la información acerca de las *dimensiones esenciales* dentro de un proceso de desarrollo de software. Son, según lo que establecen las prácticas basadas en procesos ágiles, ejemplos de radiadores de información²⁷. Contar con dicha información actualizada posibilita la realizar un seguimiento preciso del progreso en el desarrollo de una iniciativa de software, más brinda los ajustes necesarios para completar el desarrollo de la iniciativa. En el presente caso, la iniciativa está definida dentro del ámbito del actual Proyecto Final de Carrera (PFC).

Los *alphas* son aspectos de una iniciativa de software cuya evolución queremos comprender, monitorear, dirigir y controlar. Estos están relacionados al trabajo a realizar como parte del ciclo de vida para poder cumplir con los objetivos ligados al desarrollo de una iniciativa. Cada *alpha* posee un estado particular, dentro de un universo de varios posibles definidos según cada *alpha*. Ese estado permite conocer el nivel de progresión de los *alphas* dentro del ciclo de vida y así evaluar el nivel de salud de una iniciativa de software.

Bajo el enfoque propuesto por Essence, los *alphas* son las cuestiones más vitales a las cuales se les debe de prestar atención para poder conocer qué tan exitosa está resultando ser la ejecución del desarrollo de una iniciativa de software en particular.

Originalmente en el SEMAT Kernel se definen siete *alphas*. Si bien un equipo de desarrollo puede adoptar Essence y utilizar los *alphas* definidos en el SEMAT Kernel, también está dentro de las posibilidades descritas por Essence la de incluir *alphas* adicionales personalizados según una iniciativa en particular o un proceso de desarrollo en particular; lo cual transforma a Essence en un framework extensible. En la gestión del presente PFC se utilizan sólo los siete *alphas* originales de Essence.

Para hacer uso de Essence, el equipo procedió a generar copias físicas de las tarjetas utilizadas para utilizar el framework, a fin de emular el proceso detallado en Jacobson et al. (2015). Debido a la falta de traducciones fiables al español de los conceptos y las denominaciones utilizadas en Essence y en el SEMAT Kernel, el equipo decidió, junto con los conceptos en inglés original, incluir traducciones aproximadas en español de su autoría. Se incluye, para cada concepto relevante, la traducción y el concepto en su inglés original, a fin de evitar ambigüedades.

²⁷ [What is an Information Radiator? | Agile Alliance](#)

Alphas definidos por el SEMAT Kernel

Los *alphas* pueden representarse en una colección de tarjetas. Cada uno posee un número de estados, con cada *alpha* en un estado definido al comienzo del desarrollo. Luego, si corresponde, es actualizado por otro estado superior al finalizar cada iteración de desarrollo. Cada estado, para ser alcanzado, debe cumplir con una serie de requisitos, a los cuales llamaremos elementos. Los elementos están especificados en una checklist, la cual puede visualizarse en las imágenes adjuntas que ilustran cada tarjeta de estado de los diversos *alphas*.

Al comienzo del desarrollo se debe asignar un estado inicial a cada uno de los *alphas*, evaluando, a medida que se avanza sobre el desarrollo del proyecto, cuál es el estado alcanzado. Para el caso de nuestro proceso de desarrollo esto es parte de las retrospectivas que forman parte del Principio 3 de Kanban, como se describió anteriormente en el capítulo “Proceso de Desarrollo”. En cada retrospectiva se realiza la evaluación de los elementos que cumple el proyecto actual en el estado actual y sus posteriores, determinando el estado como alcanzado si todos estos están cubiertos.

Los *alphas* están agrupados en tres áreas, según su naturaleza y cómo se relacionan entre sí: Clientes, Solución e Iniciativa. A continuación, se detalla cada *alpha*, cada área que los contiene y cómo se relacionan entre sí, más las tarjetas que les representan gráficamente. La descripción completa, junto con este diagrama, más el detalle de cada estado y cada checklist dentro de los *alphas* puede encontrarse en la guía de referencia rápida del SEMAT²⁸.

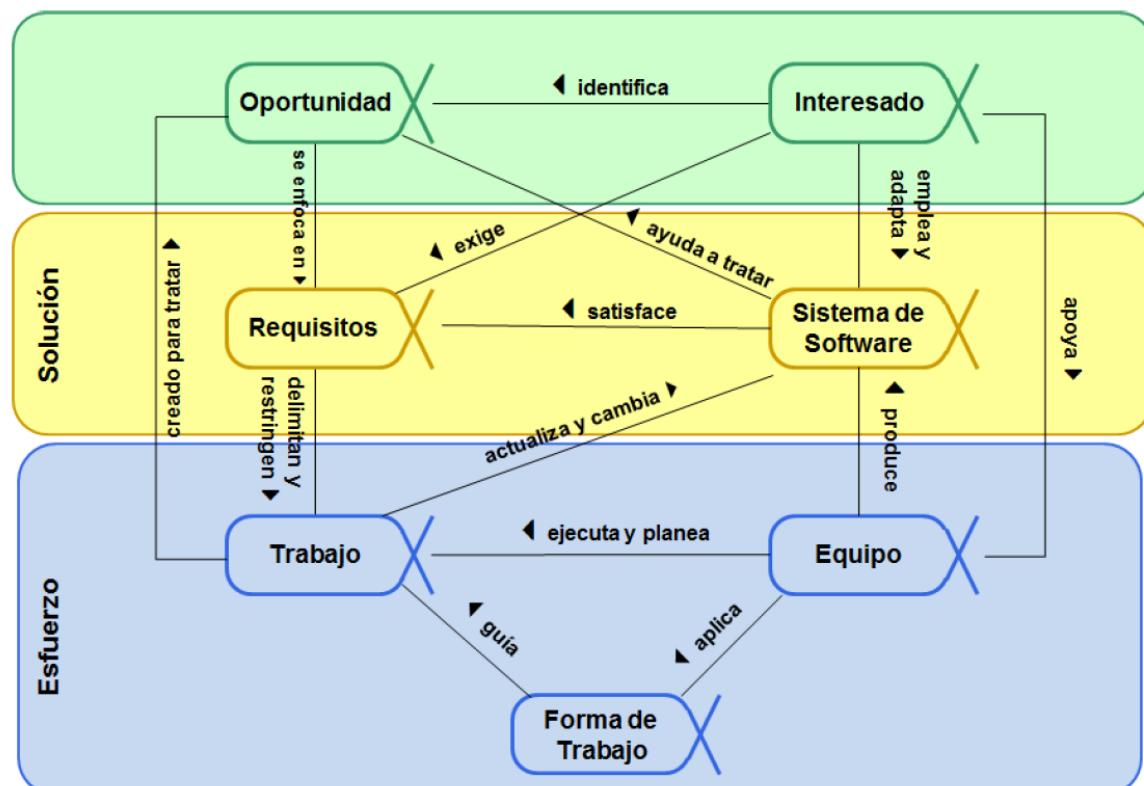


Figura 19 – Detalle de cómo los *alphas* de las distintas áreas se relacionan entre sí.

²⁸ SEMAT Quick Reference Guide: <http://semat.org/quick-reference-guide>

Área Clientes

En el área **Clientes**, el equipo necesita comprender las necesidades de los *stakeholders* y la **oportunidad** que necesita ser abordada. Las tarjetas o cartas que representan a los *alphas* del área Clientes están representadas por un color verde claro o aguamarina.

Alpha "Stakeholders"

Representa a las personas, grupos u organizaciones que afectan o son afectadas por un sistema de software. Los stakeholders o interesados proveen la **oportunidad** y son el origen de los **requerimientos** que dan lugar al desarrollo del sistema.

Los stakeholders usan y consumen el sistema de software, financian su desarrollo y están involucrados de manera activa a lo largo de la duración de la iniciativa de software. Cada parte interesada que representa a un stakeholder puede poseer representantes que colaboran con el equipo de desarrollo para alcanzar acuerdos hacia un sistema aceptable. Los inputs y el feedback provisto por los stakeholders resultan cruciales para dar forma a la iniciativa de software y al sistema de software resultante. Por definición, se incluye a los miembros del equipo de desarrollo como stakeholders.

Los estados de este *alpha* permiten evaluar el grado de evolución de la relación de los stakeholders con el equipo de desarrollo y con el producto en desarrollo.

Este *alpha* posee seis estados referidos a los Stakeholders:

- 1) Reconocidos (Recognized)
- 2) Representados (Represented)
- 3) Involucrados (Involved)
- 4) En acuerdo (In Agreement)
- 5) Satisfechos para despliegue del software (Satisfied for Deployment)
- 6) Satisfechos en el uso del software (Satisfied in Use)

 Stakeholders



Recognized	The people, groups, or organizations who affect or are affected by a software system.
Represented	The stakeholders: <ul style="list-style-type: none"> Provide the opportunity and are the source of the requirements Use and consume the software system Fund the development of the software system Actively represent the groups and organizations affected by the software system Are actively involved all the way through the endeavor Have representatives that collaborate with the team to reach agreement on an acceptable system
Involved	
In Agreement	
Satisfied for Deployment	
Satisfied in Use	

SEM|T

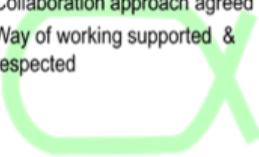
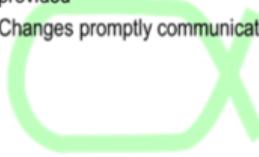
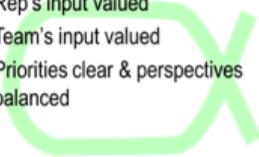
Stakeholders	Stakeholders	Stakeholders
Recognized	Represented	Involved
<input type="checkbox"/> Stakeholder groups identified <input type="checkbox"/> Key stakeholder groups represented <input type="checkbox"/> Responsibilities defined 	<input type="checkbox"/> Responsibilities agreed <input type="checkbox"/> Representatives authorized <input type="checkbox"/> Collaboration approach agreed <input type="checkbox"/> Way of working supported & respected 	<input type="checkbox"/> Representatives assist the team <input type="checkbox"/> Timely feedback and decisions provided <input type="checkbox"/> Changes promptly communicated 
SEM T 1 / 6 	SEM T 2 / 6 	SEM T 3 / 6 
Stakeholders	Stakeholders	Stakeholders
In Agreement	Satisfied for Deployment	Satisfied in Use
<input type="checkbox"/> Minimal expectations agreed <input type="checkbox"/> Rep's happy with their involvement <input type="checkbox"/> Rep's input valued <input type="checkbox"/> Team's input valued <input type="checkbox"/> Priorities clear & perspectives balanced 	<input type="checkbox"/> Stakeholder feedback provided <input type="checkbox"/> System ready for deployment 	<input type="checkbox"/> Feedback on system use available <input type="checkbox"/> System meets expectations 
SEM T 4 / 6 	SEM T 5 / 6 	SEM T 6 / 6 

Figura 20 – Detalle de las tarjetas de estado del *alpha Stakeholders*, incluyendo las checklists de elementos de cada uno.

Alpha “Oportunidad”

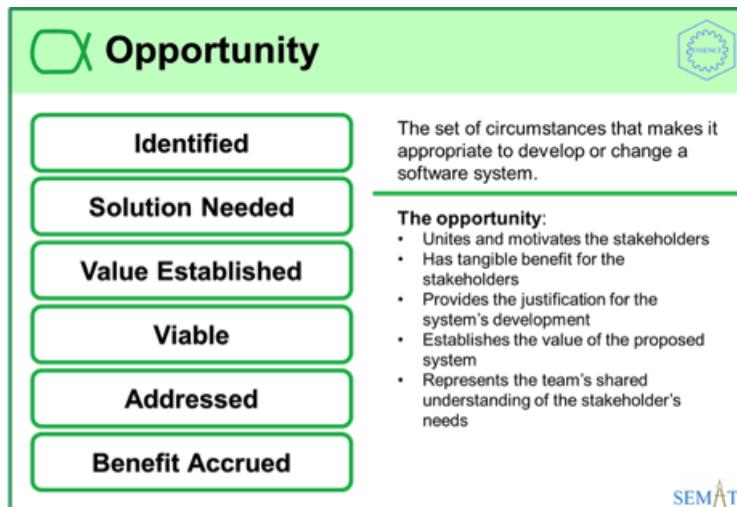
Representa el conjunto de circunstancias que vuelve apropiado el desarrollo o la introducción de modificaciones en un sistema de software.

La oportunidad articula la razón y provee la motivación para la creación de un nuevo **sistema de software** o la modificación de uno ya existente. Este *alpha* representa el entendimiento compartido de las necesidades de los **stakeholders** y sirve de base para los **requerimientos** para un nuevo **sistema de software**, justificando su desarrollo.

Mediante el análisis y la comprensión de la oportunidad es posible identificar el valor y el resultado deseado que los stakeholders esperan obtener a partir de la puesta en funcionamiento y el uso del sistema de software, sea de forma aislada o como parte de una solución técnica de mayor escala.

Este *alpha* posee seis estados posibles, referidas a la oportunidad:

1. Identificada – Identified
2. Solución requerida – Solution Needed
3. Valor establecido – Value Established
4. Viable – Viable
5. Abordadas – Addressed
6. Beneficio obtenido – Benefit Accrued



<p>Opportunity</p> <p>Identified</p> <p> <input type="checkbox"/> Idea behind opportunity identified <input type="checkbox"/> At least one investing stakeholder interested <input type="checkbox"/> Other stakeholders identified </p> <p>1 / 6</p>	<p>Opportunity</p> <p>Solution Needed</p> <p> <input type="checkbox"/> Solution identified <input type="checkbox"/> Stakeholders' needs established <input type="checkbox"/> Problems and root causes identified <input type="checkbox"/> Need for a solution confirmed <input type="checkbox"/> At least one solution proposed </p> <p>2 / 6</p>	<p>Opportunity</p> <p>Value Established</p> <p> <input type="checkbox"/> Opportunity value quantified <input type="checkbox"/> Solution impact understood <input type="checkbox"/> System value understood <input type="checkbox"/> Success criteria clear <input type="checkbox"/> Outcomes clear and quantified </p> <p>3 / 6</p>
<p>Opportunity</p> <p>Viable</p> <p> <input type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input type="checkbox"/> Reasons to develop solution understood <input type="checkbox"/> Pursuit viable </p> <p>4 / 6</p>	<p>Opportunity</p> <p>Addressed</p> <p> <input type="checkbox"/> Opportunity addressed <input type="checkbox"/> Solution worth deploying <input type="checkbox"/> Stakeholders satisfied </p> <p>5 / 6</p>	<p>Opportunity</p> <p>Benefit Accrued</p> <p> <input type="checkbox"/> Solution accrues benefits <input type="checkbox"/> ROI acceptable </p> <p>6 / 6</p>

Figura 21 – Detalle de las tarjetas de estado del alpha Oportunidad, incluyendo las checklists de cada uno.

Área Solución

En el área **Solución**, el equipo necesita establecer un entendimiento compartido de los requerimientos, y luego implementar, construir, probar, desplegar y brindar soporte a un **sistema de software** que aborde los mismos. Las tarjetas o cartas que representan a los *alphas* del área Clientes están representadas por un color amarillo claro.

Alpha “Requerimientos”

Representa lo que el sistema debe hacer para aprovechar la **oportunidad** y satisfacer las necesidades de los **stakeholders**.

Es crucial descubrir y comprender lo que se requiere del sistema de software, compartir el conocimiento generado con los stakeholders y los miembros del equipo y utilizarlo como fuerza conductora para llevar adelante el desarrollo y la generación de pruebas de un nuevo **sistema de software**.

Los requerimientos pueden ser capturados como un conjunto de ítems requeridos. Estos pueden ser comunicados y registrados de formas variadas y en diversos niveles de detalle. Es importante que el estado en general de los requerimientos es comprendido, como así también el estado de los ítems requeridos individuales.

Entre otras cosas, la comprensión de los requerimientos ayudará a establecer cuándo un sistema de software se encuentra concluido y a determinar si un requisito individual se encuentra dimensionado en alcance y complejidad.

Este *alpha* posee seis estados, referidos a los Requerimientos:

1. Ideados – Conceived
2. Delimitados – Bounded
3. Coherentes – Coherent
4. Aceptables – Acceptable
5. Abordados – Addressed
6. Cumplidos – Fulfilled



Requirements

- Conceived**
- Bounded**
- Coherent**
- Acceptable**
- Addressed**
- Fulfilled**

What the software system must do to address the opportunity and satisfy the stakeholders.

The requirements:

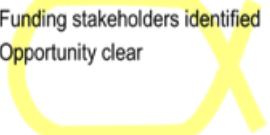
- Establish a shared understanding of what the software system must do
- Communicate the intent of the software system to be produced
- Define the capabilities, services and qualities that the stakeholders desire from the system
- Are organized to allow the scope of the software system to be managed
- Drive the development and testing of the system



Requirements

Conceived

- Stakeholders agree system is to be produced
- Users identified
- Funding stakeholders identified
- Opportunity clear

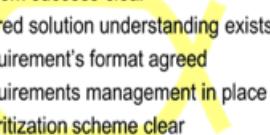


1 / 6

Requirements

Bounded

- Development stakeholders identified
- System purpose agreed
- System success clear
- Shared solution understanding exists
- Requirement's format agreed
- Requirements management in place
- Prioritization scheme clear
- Constraints identified & considered
- Assumptions clear

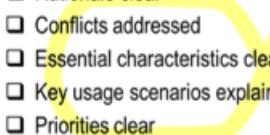


2 / 6

Requirements

Coherent

- Requirements shared
- Requirements' origin clear
- Rationale clear
- Conflicts addressed
- Essential characteristics clear
- Key usage scenarios explained
- Priorities clear
- Impact understood
- Team knows & agrees on what to deliver

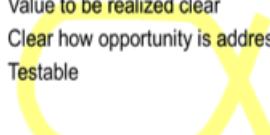


3 / 6

Requirements

Acceptable

- Acceptable solution described
- Change under control
- Value to be realized clear
- Clear how opportunity is addressed
- Testable

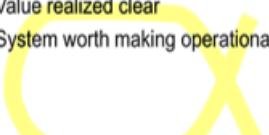


4 / 6

Requirements

Addressed

- Enough addressed to be acceptable
- Requirements and system match
- Value realized clear
- System worth making operational

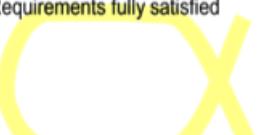


5 / 6

Requirements

Fulfilled

- Stakeholders accept requirements
- No hindering requirements
- Requirements fully satisfied



6 / 6

Figura 22 – Detalle de las tarjetas de estado del *alpha* Requerimientos, incluyendo las checklists de cada uno.

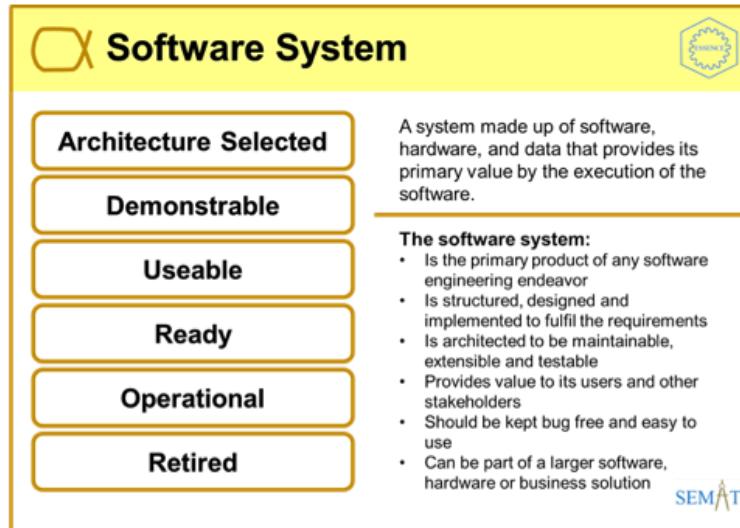
Alpha “Sistema de Software”

Representa a un sistema compuesto por un software, hardware y datos que provee su valor primario mediante la ejecución de software. Es el producto principal de cualquier iniciativa de desarrollo; un sistema de software que puede ser parte de uno mayor, hardware, un negocio o un sistema social.

Se utilizó el término sistema de software en vez de software debido a que la ingeniería de software produce artefactos más complejos que una pieza de software. Aunque el valor provenga del software y de la ejecución y uso de este, un software funcional depende de la combinación de software, hardware y los datos necesarios para cumplir con los **requerimientos**.

Este *alpha* posee seis estados, referidos al Sistema de Software en desarrollo:

1. Arquitectura del sistema Elegida – Architecture Selected
2. Sistema Demostrable – Demonstrable
3. Sistema Utilizable – Useable
4. Sistema Listo – Ready
5. Sistema Operacional – Operational
6. Sistema Retirado – Retired



Software System

Architecture Selected

- Architecture selection criteria agreed
- HW platforms identified
- Technologies selected
- System boundary known
- Decisions on system organization made
- Buy, build, reuse decisions made
- Key technical risks agreed to

SEM/IT 1 / 6

Demonstrable

- Key architectural characteristics demonstrated
- System exercised & performance measured
- Critical HW configurations demonstrated
- Critical interfaces demonstrated
- Integration with environment demonstrated
- Architecture accepted as fit-for-purpose

SEM/IT 2 / 6

Usable

- System can be operated
- System functionality tested
- System performance acceptable
- Defect levels acceptable
- System fully documented
- Release content known
- Added value clear

SEM/IT 3 / 6

Software System

Ready

- User documentation available
- System accepted as fit-for-purpose
- Stakeholders want the system
- Operational support in place

SEM/IT 4 / 6

Operational

- System available for use
- System live
- Agreed service levels supported

SEM/IT 5 / 6

Retired

- Replaced or discontinued
- No longer supported
- No authorized users
- Updates stopped

SEM/IT 6 / 6

Figura 23 – Detalle de las tarjetas de estado del *alpha Sistema de Software*, incluyendo las checklists de cada uno.

Área Iniciativa

En el área **iniciativa**, el **equipo** y su **way of working** deben tomar lugar conjuntamente con el **trabajo** que debe de ser realizado.

Alpha “Equipo”

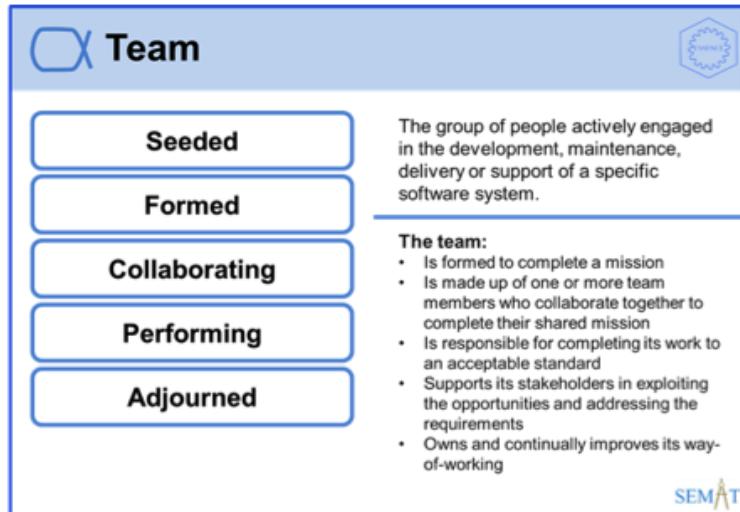
Representa a un grupo de personas activamente involucradas en el desarrollo, mantenimiento o soporte de un **sistema de software** específico. Uno o más equipos planifican y ejecutan el trabajo necesario para crear, actualizar, cambiar o retirar el sistema de software.

La ingeniería de software es una actividad que se desarrolla colaborativamente en equipo, y que requiere la aplicación de diferentes competencias y habilidades. Para poder alcanzar un buen desempeño, los miembros deben pensar acerca de cómo mejorar la manera en la que trabajan conjuntamente, teniendo en cuenta su potencial y la efectividad mediante la cual pueden lograr su misión.

Un equipo requiere de al menos dos personas, aunque la guía provista por el *alpha* de Equipo también puede resultar de utilidad para ayudar a individuos en su misión de desarrollar software. De manera excepcional, el **trabajo** puede estar asignado a un individuo que crea software puramente para su uso o entretenimiento.

Este *alpha* posee cinco estados, referidos al Equipo:

1. Preseleccionado – Seeded
2. Formado – Formed
3. Colaborando – Collaborating
4. Performando – Performing
5. Disuelto – Adjourned



Team State	Checklist Items	Card Number
Seeded	<input type="checkbox"/> Mission defined <input type="checkbox"/> Constraints known and defined <input type="checkbox"/> Growth mechanisms in place <input type="checkbox"/> Composition defined <input type="checkbox"/> Responsibilities outlined <input type="checkbox"/> Required commitment level clear <input type="checkbox"/> Required competencies identified <input type="checkbox"/> Size determined <input type="checkbox"/> Governance rules defined <input type="checkbox"/> Leadership model selected	1 / 5
Formed	<input type="checkbox"/> Individual responsibilities accepted and aligned to competencies <input type="checkbox"/> Enough members recruited <input type="checkbox"/> Roles understood <input type="checkbox"/> How to work understood <input type="checkbox"/> Members introduced <input type="checkbox"/> Members accepting work <input type="checkbox"/> External collaborators identified <input type="checkbox"/> Communication mechanisms defined <input type="checkbox"/> Members commit to team	2 / 5
Collaborating	<input type="checkbox"/> Works as one unit <input type="checkbox"/> Communication open and honest <input type="checkbox"/> Focused on mission <input type="checkbox"/> Members know each other	3 / 5
Performing	<input type="checkbox"/> Consistently meeting commitments <input type="checkbox"/> Continuously adapting to change <input type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input type="checkbox"/> Waste continuously eliminated	4 / 5
Adjourned	<input type="checkbox"/> Responsibilities fulfilled <input type="checkbox"/> Members available to other teams <input type="checkbox"/> Mission concluded	5 / 5

Figura 24 – Detalle de las tarjetas de estado del **alpha Equipo**, incluyendo las checklists de cada uno.

Alpha “Trabajo”

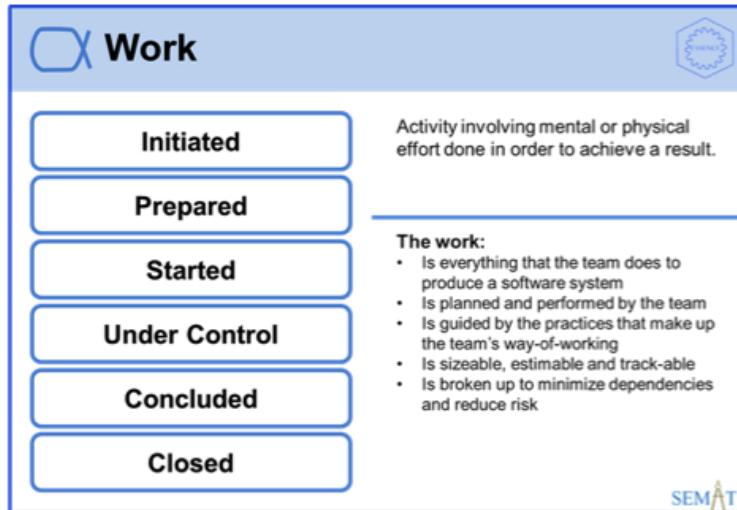
Es una actividad que involucra esfuerzo físico o mental realizado para lograr un resultado. En el contexto de la ingeniería de software, el trabajo es todo lo que el equipo realiza para lograr las metas de producir un sistema de software que corresponda los requerimientos y aproveche la oportunidad que han sido establecidos por los stakeholders. Se guía a través de las prácticas que, en conjunto, generan el **way of working** del **equipo**.

La habilidad de los miembros de un equipo de coordinar, organizar, estimar, completar y compartir su trabajo tiene un profundo efecto en lograr los objetivos de una iniciativa de software y en proveer de valor a los stakeholders. Dichos miembros requieren comprender cómo llevar adelante su trabajo y cómo reconocer cuándo el trabajo está siendo realizado de forma efectiva.

Este *alpha* posee cinco estados, referidos al Trabajo:

1. Iniciado – Initiated
2. Preparado – Prepared
3. Comenzado – Started
4. Bajo control – Under Control
5. Concluido – Concluded
6. Finalizado – Closed

Si bien la terminología utilizada para los dos últimos estados puede resultar confusa, debe notarse que el último estado contempla que se han terminado todas las tareas, tanto de ingeniería de software como administrativas, y que además se ha procedido a realizar una evaluación cualitativa y cuantitativa respecto de los resultados del trabajo y también respecto de lo aprendido por el equipo en base al trabajo realizado.



Work

Initiated

- Required result clear
- Constraints clear
- Funding stakeholders known
- Initiator identified
- Accepting stakeholders known
- Source of funding clear
- Priority clear

SEMAT 1 / 6

Work

Prepared

- Commitment made
- Cost and effort estimated
- Resource availability understood
- Risk exposure understood
- Acceptance criteria established
- Sufficiently broken down to start
- Tasks identified and prioritized
- Credible plan in place
- At least one team member ready
- Integration points defined

SEMAT 2 / 6

Work

Started

- Development started
- Progress monitored
- Definition of done in place
- Tasks being progressed

SEMAT 3 / 6

Work

Under Control

- Tasks being completed
- Unplanned work under control
- Risks under control
- Estimates revised to reflect performance
- Progress measured
- Re-work under control
- Commitments consistently met

SEMAT 4 / 6

Work

Concluded

- Only admin tasks left
- Results achieved
- Resulting system accepted

SEMAT 5 / 6

Work

Closed

- Lessons learned
- Metrics available
- Everything archived
- Budget reconciled & closed
- Team released
- No outstanding, uncompleted tasks

SEMAT 6 / 6

Figura 25 – Detalle de las tarjetas de estado del alpha Equipo, incluyendo las checklists de cada uno.

Alpha “Way of Working”

Representa al conjunto de prácticas y herramientas específicas del equipo, el cual es utilizado por éste para guiar y brindar soporte a su trabajo. El equipo mejora continuamente su way of working conjuntamente con la mejora de su comprensión de su misión como equipo y el ambiente de trabajo que sus miembros comparten.

Establecer un way of working ayuda al equipo a ser efectivo en la ejecución del **trabajo**. El way of working continuamente es inspeccionado, adaptado y mejorado, mientras el desarrollo del sistema de software es llevado adelante.

El equipo de desarrollo es el *dueño* del way of working, y es el equipo quien lo pone en práctica y lo define. El way of working refleja las políticas y los estándares adoptados por el equipo. La generación de un way of working consolidado reduce el riesgo y ayuda a reducir pérdidas en el tiempo de trabajo.

Este *alpha* posee cinco estados, referidos al Way of Working:

1. Principios establecidos – Principles Established
2. Fundamentos establecidos – Foundation Established
3. En uso – In Use
4. En lugar – In Place
5. Funcionando bien – Working Well
6. Retirado – Retired

Way of Working



Principles Established

The tailored set of practices and tools used by the team members to guide and support their work.

Foundation Established

The way of working:

- Helps the team be effective and work well together
- Evolves as the team understands its mission and environment, and undertakes its work
- Is continually inspected, adapted and improved
- Is owned and agreed by the team
- Reflects organizational policies and standards
- Reduces risk and helps to eliminate waste

In Use

In Place

Working Well

Retired



Way of Working

Principles Established

- Team actively support principles
- Stakeholders agree with principles
- Tool needs agreed
- Approach recommended
- Operational context understood
- Practice & tool constraints known

Way of Working

Foundation Established

- Key practices & tools selected
- Practices needed to start work agreed
- Non-negotiable practices & tools identified
- Gaps between available and needed way-of-working understood
- Gaps in capability understood
- Integrated way of working available

Way of Working

In Use

- Practices & tools in use
- Regularly inspected
- Adapted to context
- Supported by team
- Feedback mechanisms in place
- Practices & tools support collaboration

Way of Working

In Place

- Used by whole team
- Accessible to whole team
- Inspected and adapted by whole team

Way of Working

Working Well

- Predictable progresses being made
- Practices naturally applied
- Tools naturally support way-of-working
- Continually tuned

Way of Working

Retired

- No longer in use
- Lessons learned shared



4 / 6



5 / 6



6 / 6

Figura 26 – Detalle de las tarjetas de estado del alpha Way of Working, incluyendo las checklists de cada uno.

Utilización del SEMAT Kernel para la gestión del proyecto

La planificación del presente Proyecto Final de Carrera establece que el desarrollo tendrá, dentro del ciclo de vida, **seis** iteraciones. Como parte del ciclo de vida del proyecto, al final de cada iteración se pautó el despliegue y demostración de un entregable para revisión de los stakeholders.

Haciendo uso de la planificación trazada, el equipo decidió utilizar estas iteraciones para, siguiendo los principios establecidos por Kanban, proceder a hacer revisiones sobre el trabajo y el way of working, utilizando para ello ceremonias de retrospectiva. Las métricas brindadas por Essence, en forma de *alphas*, fueron usadas al comienzo de cada iteración para evaluar el estado de salud de la iniciativa de software.

Inicialmente, para dar comienzo a lo concerniente a la gestión del proyecto, se ejecutó una *iteración 0*, a fin de determinar cuál es el estado de los *alphas* antes del comienzo formal de la ejecución del proyecto. Se tomó esta decisión debido a que: (1) el equipo previamente ya se encontraba trabajando y colaborando en iniciativas anteriores que sirven de base al proyecto actual y (2) el equipo posee información útil relativa a cómo llevar adelante la ejecución del proyecto debido a la documentación de planificación generada en la etapa previa a la ejecución del proyecto. Por lo tanto, al principio del proyecto los *alphas* no necesariamente estaban en su estado inicial.

Para realizar la evaluación regular de la gestión del proyecto, con el marco brindado por la finalización de cada iteración, se procedió a realizar las ceremonias de retrospectiva. En ese momento, como parte de las reuniones, se llevaron adelante evaluaciones sobre la gestión del proyecto, chequeando el estado alcanzado en cada uno de los *alphas* actuales, revisando si existieron cambios de estado y realizando para ello la correspondiente revisión de las *checklists* de cada uno.

En estas ceremonias, se procedió a realizar una revisión de los *checklists* de cada uno de los *alphas* y sus estados particulares, actualizando los diversos estados alcanzados cuando correspondiera y analizando cuándo se detectaba que no había evolución en determinados estados, a fin de determinar cuáles eran los impedimentos que estaban retrasando el desarrollo y la evolución de estado de esos *alphas*. Para la evaluación de cada punto de las *checklists* de los estados de los *alphas*, se recurrió a las definiciones descriptas por Jacobson et al. (2013).

A continuación, se procede a explicitar cuál es el estado inicial de los *alphas* como parte de la *iteración 0*, para luego comenzar a relatar cuáles fueron las transiciones en los estados de los *alphas* mientras el desarrollo del proyecto fue avanzando. Conjuntamente con esta definición, se incluyen las justificaciones respecto de cómo se llega a la conclusión respecto del estado inicial y los objetivos dentro de cada estado que se cumplen en las áreas Clientes, Software e Iniciativa.

Para ilustrar la manera en la cual los estados alcanzados para cada *alpha* como parte de cada iteración, se procederá a incluir cuáles son las tarjetas de cada estado alcanzado, relatando cómo se ha cumplido cada ítem de las diversas *checklists*. De esta manera, se

busca detallar en la sección siguiente cómo el equipo procedió a ejecutar la evaluación de los *alphas*, emulando el uso de las cartas de Essence adquiridas por el equipo para utilizar el framework en las reuniones de retrospectiva.

Posteriormente al detalle de lo evaluado en la *iteración 0*, se procede a detallar el logro de los alcances de los determinados estados de los *alphas* a través del desarrollo de la iniciativa, marcando estados siguientes como objetivo de cada una de las iteraciones. Finalmente, se procederá a detallar la retrospectiva final orientada a proveer al equipo de *insights* y conclusiones acerca de cómo se llevó adelante el proyecto desde una perspectiva general.

Iteración 0

Evaluación de estados en la iteración 0

Al comenzar la etapa de ejecución²⁹ del presente Proyecto Final de Carrera (PFC), el equipo a cargo ya contaba con experiencia previa a causa del desarrollo de los mencionados prototipos de software. El desarrollo conjunto de estas iniciativas sirvió de base para lo concerniente a la provisión de la base de código, los procedimientos de trabajo, la experiencia en la tecnología y el abordaje de la oportunidad para el desarrollo del presente proyecto.

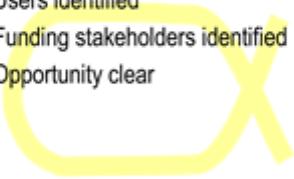
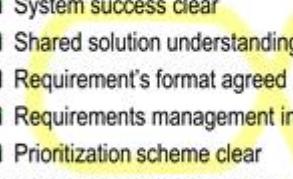
Además de esta experiencia, la evaluación realizada para la *iteración 0* considera el conocimiento y la información generada a partir de la entrega del documento de planificación del PFC, el cual fue revisado y aprobado por el director del PFC con anterioridad al comienzo de la etapa de ejecución.

Tomando en cuenta estas consideraciones, sobre las cuales se vuelve en detalle en la evaluación particular de los elementos alcanzados dentro de cada estado de los *alphas*, se procede a detallar cuáles fueron, a consideración del equipo y su experiencia, los estados alcanzados de cada uno de los *alphas* previamente al comienzo de la etapa de ejecución del PFC.

²⁹ La “etapa de planificación” refiere a aquella que culminó con la entrega del Plan de Proyecto Final de Carrera, la cual es previa a la “etapa de ejecución”. Definimos a esta última es la que atañe al diseño, construcción, prueba e implementación del sistema de software.

Alpha “Requerimientos”

El equipo considera que el *alpha* de Requerimientos, al comienzo de la iniciativa, se encontraba en el estado 2 de 6, que refiere a los requerimientos como “**Acotados**” (**Bounded**).

Requirements	Requirements
Conceived  <input checked="" type="checkbox"/> Stakeholders agree system is to be produced <input checked="" type="checkbox"/> Users identified <input checked="" type="checkbox"/> Funding stakeholders identified <input checked="" type="checkbox"/> Opportunity clear	Bounded  <input checked="" type="checkbox"/> Development stakeholders identified <input checked="" type="checkbox"/> System purpose agreed <input checked="" type="checkbox"/> System success clear <input checked="" type="checkbox"/> Shared solution understanding exists <input type="checkbox"/> Requirement's format agreed <input checked="" type="checkbox"/> Requirements management in place <input checked="" type="checkbox"/> Prioritization scheme clear <input type="checkbox"/> Constraints identified & considered <input type="checkbox"/> Assumptions clear
SEMAT 1 / 6	SEMAT 2 / 6

Estado “Concebidos”

El primer estado de este alpha, “**Concebidos**” (**Conceived**), tiene cuatro elementos para ser alcanzado. El equipo de desarrollo consideró alcanzados los cuatro elementos al inicio de la etapa de ejecución. Este estado implica que hay consenso respecto de la necesidad del desarrollo de un nuevo sistema de software por parte de los stakeholders.

“Stakeholders agree system is to be produced” (Los stakeholders acuerdan en que el sistema debe de ser producido): Los stakeholders estaban de acuerdo con el desarrollo del sistema, dada la manifestación de la necesidad de resolver un problema específico (la gestión de un congreso académico), mediante una solución de software que motivase a llevar adelante la iniciativa.

“Users identified” (Usuarios identificados): Los usuarios se encontraban identificados como asistentes al evento ENIEF 2019, y usarían la solución de software para facilitar la experiencia de su asistencia al ENIEF 2019.

“Funding stakeholders identified” (Stakeholders que financiarán el proyecto identificado): Los stakeholders que financiaron el presente proyecto estaban identificados; los organizadores del evento ENIEF 2019. La organización encargada de llevarlo adelante es el CIMEC-CONICET (Centro de Investigación en Métodos Computacionales; Consejo

Nacional de Investigaciones Científicas y Técnicas) juntamente con la AMCA (Asociación Argentina de Mecánica Computacional).

Si bien el financiamiento no estaba completamente garantizado, dado que la presente propuesta competía con otras postulaciones para oficiar de aplicativo de gestión oficial del evento, el equipo consideró que esta situación bastaba para asumir que existía una posibilidad de financiamiento para el proyecto.

“Opportunity clear” (Oportunidad clara): La oportunidad, comprendida al momento de la evaluación de este elemento por el equipo de desarrollo y por los stakeholders, se refiere a poder hacer uso de una solución de software para hacer más eficiente y efectiva la gestión de un congreso académico y elevar la calidad de interacción de los asistentes a dicho evento. Es, en resumen, la oportunidad que el nuevo sistema abordará.

Estado “Acotados”

El segundo estado de este *alpha*, Requerimientos “**Acotados**” (**Bounded**), refiere a la comprensión del propósito y el contexto dentro del cual el sistema debe de operar. Al comienzo de la ejecución, este estado se alcanzó de forma parcial. Se detalla a continuación cuáles fueron los elementos logrados y los no logrados.

Logrados

“Development stakeholders identified” (Stakeholders desarrolladores identificados): Los stakeholders que estuvieron involucrados en el desarrollo del nuevo sistema se encontraban identificados, siendo este el equipo los autores del presente proyecto: Ramiro Olivencia y Federico Albertengo. Como parte de los stakeholders destinatarios de la iniciativa, el Dr. Pablo Kler fue quien ofició de intermediario entre la organización de ENIEF y el equipo, siendo consultado como experto del dominio del problema para el desarrollo de la iniciativa.

“System purpose agreed” (Consenso sobre el propósito del sistema): El propósito del sistema era claro, consistiendo en servir de plataforma para la gestión digital del congreso ENIEF 2019, en base a los requerimientos ya especificados en la planificación del presente proyecto.

“System success clear” (Visión de éxito del sistema identificada): Hubo consenso respecto de cómo el sistema debería operar para que su desarrollo exitoso. El hecho de poder cumplir con los requerimientos definidos en la planificación en tiempo y forma, juntamente con un sólido abordaje de los riesgos técnicos definidos en la planificación, fueron considerados como puntos centrales respecto de qué representaría un éxito en el desarrollo del proyecto.

“Shared solution understanding exists” (Stakeholders poseen un conocimiento compartido entre lo que representa la solución de software): El equipo de desarrollo y los clientes del proyecto ya poseían en esta etapa un conocimiento compartido acerca de qué

valor aportaría la implementación de una solución de software que aborde los requerimientos especificados.

“Requirements management in place” (Existencia de una forma de gestión de los requerimientos): Debido a la existencia de un proceso de desarrollo de software ya determinado, el equipo ya contaba con un mecanismo para poder gestionar los requerimientos de software, fuesen estos los existentes en la etapa de planificación como también los no planificados.

“Prioritization scheme clear” (Esquema de priorización claro): Al igual que en el punto anterior, el hecho de poseer una lista de requerimientos cuya prioridad ya se encuentra preestablecida en la planificación, juntamente con la posibilidad de priorización posterior que nos ofrece el proceso de desarrollo basado en Kanban, nos brinda el soporte suficiente para asumir a este elemento como alcanzado.

No logrados

“Requirement’s format agreed” (Consenso en el formato de los requerimientos): Durante la *iteración 0* se consideró que no existía una forma consensuada o definida respecto a cómo tratar los nuevos requerimientos que surgieran durante las posteriores iteraciones del desarrollo.

“Constraints identified & considered” (Limitaciones identificadas y consideradas): Si bien el equipo hizo una evaluación de riesgos durante la etapa inicial de planificación del proyecto, se tomó a consideración la posibilidad de no haber identificado aún otras limitaciones que pueden afectar al desarrollo de la iniciativa. Es por ello que este elemento, aunque cumplido parcialmente, se deja sin marcar como alcanzado.

“Assumptions are clearly stated” (Suposiciones claramente declaradas): El equipo considera que no se establecieron las suposiciones respecto de todo lo que se requiere que el sistema debe de hacer. Requerimientos adicionales podrían surgir en etapas posteriores, y es por ello que el equipo decidió adoptar un enfoque conservador al respecto y considerar este elemento como pendiente de resolver.

Alpha “Sistema de Software”

El equipo consideró que el alpha de Sistema de Software, al comienzo de la iniciativa, se encontraba en el estado 2 de 6, el cual refiere al sistema como “**Demostrable**” (**Demonstrable**).

Software System	Software System
Architecture Selected <ul style="list-style-type: none"><input checked="" type="checkbox"/> Architecture selection criteria agreed<input checked="" type="checkbox"/> HW platforms identified<input checked="" type="checkbox"/> Technologies selected<input checked="" type="checkbox"/> System boundary known<input checked="" type="checkbox"/> Decisions on system organization made<input checked="" type="checkbox"/> Buy, build, reuse decisions made<input checked="" type="checkbox"/> Key technical risks agreed to	Demonstrable <ul style="list-style-type: none"><input type="checkbox"/> Key architectural characteristics demonstrated<input checked="" type="checkbox"/> System exercised & performance measured<input type="checkbox"/> Critical HW configurations demonstrated<input type="checkbox"/> Critical interfaces demonstrated<input type="checkbox"/> Integration with environment demonstrated<input checked="" type="checkbox"/> Architecture accepted as fit-for-purpose
SEMAT 1 / 6	SEMAT 2 / 6

Estado “Arquitectura Seleccionada”

El primer estado de este *alpha* se refiere al sistema de software como con su **Arquitectura Seleccionada**. Esto implica que al alcanzar este estado la arquitectura del sistema de software ha sido seleccionada y se ha considerado apta para abordar tanto los riesgos técnicos principales como las limitaciones organizacionales que puedan afectar al desarrollo de la iniciativa.

“Architecture selection criteria agreed (Criterios acordados para la selección de la arquitectura)”: Con la experiencia del desarrollo de iniciativas de software previas, las cuales proveen las bases del incremento de software relacionadas al presente proyecto, el equipo de desarrollo encontró que la arquitectura de iteraciones previas de la plataforma de software se acomodaba a las necesidades del incremento y permitía el abordaje de nuevos módulos orientados a satisfacer los requerimientos del cliente.

“Hardware platforms identified (Plataformas de hardware identificadas)”: Las plataformas de software soportadas por la solución de software eran conocidas al momento de la planificación del presente proyecto. Habiendo elegido una arquitectura web cliente-servidor, las plataformas que involucrarían a la solución de software se dividían en dos grupos:

Plataforma del servidor: Servidor web Apache + Tomcat, bajo un alojamiento web en una máquina virtual de Linux, accesible vía SSH.

Plataforma de la aplicación: Terminales móviles con sistemas operativos iOS y Android, en formato para tablet o smartphone.

“Technologies selected (Tecnologías seleccionadas)”: La lista de tecnologías elegidas para el desarrollo fue especificada al momento de su planificación preliminar. Por ello, se determinó que este elemento fue alcanzado al momento de la *iteración 0*.

“System boundary known (Fronteras del sistema conocidas)”: Mediante (1) la identificación de los requerimientos y el abordaje que se decidió darle a los mismos como parte de la etapa de planificación y (2) las decisiones referidas a la arquitectura de software realizadas, el equipo consideró que las fronteras del sistema de software eran conocidas al momento de la *iteración 0*. La documentación disponible en el presente proyecto referida a diagramas de arquitectura y requerimientos sienta las bases sobre las cuales el equipo determinó cómo identificar las fronteras del sistema de software.

“Decisions on system organization made (Decisiones sobre la organización del sistema en acuerdo)”: Habiéndose identificado cómo se relacionarían las distintas partes del sistema para interactuar entre sí, en base a la arquitectura elegida y las plataformas de hardware a soportar, al momento de la *iteración 0* el equipo tenía definido cómo organizar el sistema y cómo las distintas partes se interrelacionarían para posibilitar su funcionamiento.

“Buy, build, reuse decisions made (Decisiones de compra, reuso y desarrollo realizadas)”: Este elemento se refiere a la decisión acerca de qué parte del software se construirá, cuál se comprará y qué software existente se reutilizará para el desarrollo de la iniciativa. Contando con una lista de herramientas y tecnologías (todas disponible gratuitamente o bajo licencia educativa) y con la base de Código de los prototipos preliminares, al momento de la planificación ya era conocida cuál eran los requerimientos necesarios para abordar la construcción de nuevo software. Por tal motivo, se consideró a este elemento como alcanzado al momento de la *iteración 0*.

“Key technical risks agreed to (Acuerdo en los riesgos técnicos principales)”: La etapa de planificación preliminar exigía una identificación y una gestión de los riesgos identificados para el desarrollo del presente proyecto. Contando con esta documentación y habiendo la misma sido aprobada por los stakeholders al momento de la planificación preliminar, el equipo consideró a este elemento como alcanzado al momento de la *iteración 0*.

Estado “Demostrable”

El segundo estado de este *alpha* se refiere al sistema de software como **Demostrable (Demonstrable)**. Esto implica que existe una versión del sistema de software ejecutable, la cual permite exponer que la arquitectura es adecuada para el propósito del sistema y que, además, soporta la posibilidad de poder realizar pruebas para validarla.

Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 0*. La parcialidad en el alcance se debe a que este estado del *alpha* Sistema de

Software requiere de la posibilidad de contar con ciertos aspectos demostrables del sistema que, siendo referidos a un incremento de software, no habían aún sido desarrollados al momento de la *iteración 0*.

Logrados

“Architecture accepted as fit-for-purpose (Arquitectura aceptada como apta para el propósito)”: Dado que la arquitectura seleccionada ya había sido probada en una iniciativa de software anterior, y siendo la nueva solución de software a desarrollar una aplicación que se usaría en un ambiente y contexto de características similares, el equipo y los stakeholders consideraron que la arquitectura seleccionada era apta para el propósito de la solución de software a desarrollar al momento de la *iteración 0*.

“System exercised & performance measured (Sistema en ejercicio y performance mensuradas)”: Parcialmente la arquitectura seleccionada para el sistema de software había sido probada en contextos de uso reales al momento del desarrollo de los prototipos preliminares, habiendo demostrado ser apta y performante su uso intensivo por parte de múltiples usuarios en simultáneo. De esta manera, y significando el desarrollo del presente proyecto un incremento de esa arquitectura, el equipo decidió considerar a este elemento como alcanzado al momento de la *iteración 0*.

Tomando como base estas experiencias, el equipo consideró que existía evidencia de respaldo respecto a la usabilidad y la performance medible requerida para el producto final de software.

No logrados

“Key architectural characteristics demonstrated (Características claves de la arquitectura demostradas)”: Debido a los múltiples cambios que demandaría la implementación de los nuevos requerimientos dentro de la arquitectura seleccionada para poder generar una demostración del trabajo y debido a que aún no se había comenzado con la ejecución del proyecto, no resultaba posible determinar este elemento como alcanzado.

“Critical hardware configurations demonstrated (Configuración crítica de hardware demostrada)”: Al no haber alcanzado la etapa en la cual el equipo llevaría adelante el despliegue del software construido, más el correspondiente análisis previo respecto de las necesidades de la fase de despliegue dentro de la *iteración 1*, no resultó viable determinar este elemento como alcanzado.

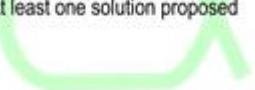
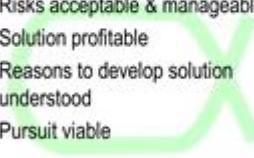
“Critical interfaces demonstrated (Interfaces críticas demostradas)”: Al no contar con un software utilizable referido a los nuevos requerimientos, y por tanto sin interfaces demostrables referidas al incremento que representa a la iniciativa del PFC, el equipo decidió considerar a este elemento como no alcanzado.

Por otra parte, el equipo sí contaba para esta etapa con interfaces demostrables respecto de componentes cuya existencia resultó previa a la ejecución del incremento.

“Integration with environment demonstrated (Integración con el ambiente demostrada)”: Debido a no contar con la posibilidad de evaluar software desplegado para la presente iniciativa, no resultó tampoco posible, para la *iteración 0*, poder demostrar la integración del software con el ambiente en el cual se ejecutaría (ni tampoco en uno similar).

Alpha “Oportunidad”

El equipo consideró que el *alpha* de Oportunidad, al inicio de la ejecución de la iniciativa, se encontraba en el estado 4 de 6, que refiere a la oportunidad como “**Viable**” (ídem para el nombre en inglés).

<p>Opportunity</p> <p>Identified</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Idea behind opportunity identified <input checked="" type="checkbox"/> At least one investing stakeholder interested <input checked="" type="checkbox"/> Other stakeholders identified  <p>SEMA/T 1 / 6</p>	<p>Opportunity</p> <p>Solution Needed</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution identified <input checked="" type="checkbox"/> Stakeholders' needs established <input checked="" type="checkbox"/> Problems and root causes identified <input checked="" type="checkbox"/> Need for a solution confirmed <input checked="" type="checkbox"/> At least one solution proposed  <p>SEMA/T 2 / 6</p>	<p>Opportunity</p> <p>Value Established</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Opportunity value quantified <input checked="" type="checkbox"/> Solution impact understood <input checked="" type="checkbox"/> System value understood <input checked="" type="checkbox"/> Success criteria clear <input checked="" type="checkbox"/> Outcomes clear and quantified  <p>SEMA/T 3 / 6</p>
<p>Opportunity</p> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable  <p>SEMA/T 4 / 6</p>		

Estado “Identificada”

El primer estado de este *alpha* se refiere a la oportunidad que motiva a una iniciativa de software como **Identificada (Identified)**. Esto implica que el equipo de desarrollo ha identificado una oportunidad de índole comercial, social o de negocios que puede ser abordada mediante el desarrollo de una solución basada en software.

“Idea behind opportunity identified” (Idea detrás de la oportunidad identificada): El equipo consideró alcanzado este elemento debido a que la idea para abordar la oportunidad

en cuestión logró ser identificada en base a la similitud del contexto con otras oportunidades similares. En otras palabras, los organizadores del ENIEF 2019 consideraron aplicable, en el contexto de su congreso académico, la solución de software aplicada en congresos anteriores en la forma de los prototipos previos que se mencionan como base del presente proyecto.

“At least one investing stakeholder interested” (Al menos un stakeholder inversor identificado): El equipo consideró como alcanzado este elemento debido a que los organizadores del ENIEF 2019 demostraron su interés en llevar adelante una solución de software en base a la oportunidad presentada.

“Other stakeholders identified (Otros stakeholders identificados)”: El equipo consideró como alcanzado este elemento debido a que no existió otro grupo adicional de stakeholders, por fuera de los miembros del equipo de desarrollo y los organizadores y asistentes del ENIEF 2019, a lo largo de la planificación y el desarrollo de la iniciativa.

Estado “Requerida de una Solución”

El segundo estado de este *alpha* se refiere a la oportunidad como **Requerida de una Solución (Solution Required)**. Esto significa que la necesidad específica de abordar la oportunidad existente mediante una solución de software no sólo *puede* ser abordada mediante una solución de software, sino que además se *confirma* esa necesidad.

“Solution identified (Solución identificada)”: El equipo consideró este elemento como alcanzado dado que la forma general de la solución, en forma de un sistema de software, ya se encontraba identificada al momento de la *iteración 0*, a lo cual se suma el conocimiento de la forma general de la solución por similitud con sistemas de software de los cuales la nueva solución heredaría funcionalidad.

“Stakeholders’ needs established (Necesidades de los stakeholders establecidas)”: El equipo consideró que, a raíz de lo especificado durante la fase de planificación, se tuvo un conocimiento concreto respecto de cuáles son las necesidades generales que los stakeholders buscaban satisfacer mediante el desarrollo de la iniciativa de software. Por lo tanto, se consideró a este elemento como alcanzado.

“Problems and root causes identified (Problemas y causas de raíz identificadas)”: El equipo consideró, dada la naturaleza del contexto, que los problemas y las causas raíz que debía de abordar la solución se conocían y se encontraban identificadas. Esto surgió de la generación de soluciones de software previas en contextos similares al que suponía la solución a desarrollar para ENIEF 2019.

“Need for a solution confirmed (Necesidad de una solución confirmada)”: El equipo consideró que en esta etapa la necesidad ya estaba confirmada, debido a las avanzadas negociaciones con el cliente respecto de llevar adelante una propuesta que satisfaga los requerimientos.

“At least one solution proposed (Al menos una solución propuesta)”: El equipo consideró que, en esta etapa, una solución posible ya había sido explicitada a los stakeholders, en forma de solución de software que representase un incremento a otras soluciones previas que el equipo desarrolló.

Estado “Valor Establecido”

El tercer estado de este *alpha* se refiere a la oportunidad como con **Valor Establecido**. Esto refiere a que se ha determinado que el valor de una solución basada en software se encuentra consensuado y que, a su vez, se comprende el impacto que conlleva, para todo los involucrados en la iniciativa, el proceder con la realización de un proyecto que busque la generación de ese valor.

“Opportunity value quantified (Valor de la oportunidad cuantificado)”: El equipo y los stakeholders consideraron, en esta etapa, que, dada la inversión de recursos materiales y esfuerzo de trabajo que el llevar adelante un congreso académico, existe la oportunidad de explorar opciones tendientes a minimizar la cantidad de inversión y esfuerzo necesarios para que el congreso pueda planificarse, organizarse y gestionarse de forma óptima invirtiendo un valor monetario menor a otras opciones ya consideradas.

“Solution impact understood (Comprensión del impacto de la solución)”: El equipo consideró alcanzado este elemento debió a que los stakeholders tuvieron una clara comprensión respecto de las ventajas que ofrecía el desarrollo de un sistema de software para la gestión del congreso académico. Además, fue comprendido cómo la puesta en funcionamiento de este sistema no sólo resultaba conveniente en términos de costo de adquisición de materiales sino también en la reducción de la cantidad de trabajo humano necesario para brindar a los asistentes una experiencia óptima durante la asistencia al congreso académico.

“System value understood (Comprensión del valor del sistema)”: El equipo consideró que este elemento se cumplió durante esta etapa, al fijar el valor de costo del desarrollo de la solución de software y que la misma era adecuada en los términos del ahorro producido en lo que respecta a la utilización de material de gestión y de referencia en formato papel durante la organización y el transcurso del congreso académico ENIEF 2019.

“Success criteria clear (Criterios de éxito claros)”: El equipo consideró que los criterios mediante los cuales el abordaje de la oportunidad a partir del desarrollo de un sistema de software resultaría un éxito se encontraban establecidos. Los requerimientos identificados durante la planificación fueron útiles para la determinación de este elemento, juntamente con el conocimiento de los stakeholders acerca de cómo la adaptación de una solución de software ya existente podía resultar útil para los requerimientos del ENIEF 2019.

“Outcomes clear and quantified (Resultados claros y cuantificados)”: El equipo consideró que los resultados que arrojaría el abordaje de la oportunidad representada se comprendían y se encontraban cuantificados. Eso implica que había entendimiento de que el desarrollo de una solución basada en software para abordar la oportunidad de gestionar de

forma más eficiente el ENIEF 2019 redundaría en resultados positivos para los stakeholders, considerando incluso la inversión en otras soluciones no basadas en software.

Estado “Viable”

El cuarto estado de este *alpha* se refiere a la oportunidad como **Viable** (ídem en inglés). Esto significa que además de estar identificado el valor, se suman a ellos una aproximación de cómo se abordará la solución de software, juntamente con la identificación de cómo se gestionarán los riesgos que implica el llevar adelante la iniciativa. Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 0*.

Logrados

“Solution outlined (Solución esbozada)”: En base a 1) las soluciones de software desarrolladas por el equipo para eventos anteriores y 2) tomando en cuenta la planificación realizada para el presente proyecto, el equipo determinó que existía, al momento de la *iteración 0*, un esbozo de la solución de software a desarrollar para la gestión del ENIEF 2019.

“Risks acceptable and manageable (Riesgos aceptables y gestionables)”: El equipo consideró que, en base a la identificación de riesgos llevada adelante como parte de la etapa de planificación, existe una comprensión y cuantificación de los riesgos inherentes. Se ha demostrado, además, que mediante la metodología adoptada en la planificación³⁰ es posible una gestión, mitigación y abordaje de los riesgos identificados.

“Reasons to develop solution understood (Comprendidas las razones que justifican desarrollar la solución)”: El equipo consideró que, para esta etapa, las razones que impulsaban a los stakeholders a requerir el desarrollo de una solución de software se encontraban ya comprendidas en claridad, dadas las múltiples ventajas de abordar la oportunidad existente y generar una gestión más eficiente y rentable del evento ENIEF 2019.

“Pursuit viable (Actividad de desarrollo viable)”: Se consideró que, para esta etapa, el desarrollo de la iniciativa se demostraba viable y dentro de las condiciones aceptables para el equipo de desarrollo. Por tanto, se aceptó seguir los pasos necesarios para materializar la iniciativa en un proyecto de software funcional.

No logrados

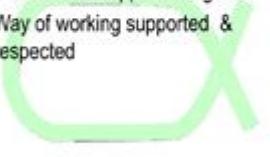
“Solution profitable (Solución rentable)”: Para esta etapa, el equipo consideró que la oportunidad abordada por la solución de software no podía ser aún considerada monetariamente beneficiosa para el equipo de desarrollo, debido a que, para lograr su financiamiento, el proyecto debía superar una etapa de selección previa, compitiendo contra otras posibles soluciones desarrolladas por otros candidatos.

³⁰ Sommerville, Ian - Software Engineering (9th edition); cap. 22.1.1

“Solution possible within constraints (Solución posible dentro de las restricciones identificadas)”: Si bien el equipo consideraba que los riesgos podían ser abordados y podían ser gestionados y evaluados, se decidió a optar por una revisión de este estado en una etapa más avanzada del proyecto, una vez que se encontrasen ya implementados algunos cambios arquitectónicos esenciales para abordar requisitos particulares de la comunicación cliente-servidor del sistema de software.

Alpha “Stakeholders”

El equipo consideró que el alpha de Stakeholders, al comienzo de la iniciativa, se encontraba en el estado 3 de 6, el cual refiere a los stakeholders como “**Involucrados**” (**Involved**).

Stakeholders	Stakeholders
Recognized	Represented
<ul style="list-style-type: none"><input checked="" type="checkbox"/> Stakeholder groups identified<input checked="" type="checkbox"/> Key stakeholder groups represented<input checked="" type="checkbox"/> Responsibilities defined 	<ul style="list-style-type: none"><input checked="" type="checkbox"/> Responsibilities agreed<input checked="" type="checkbox"/> Representatives authorized<input checked="" type="checkbox"/> Collaboration approach agreed<input checked="" type="checkbox"/> Way of working supported & respected 
 1 / 6 	 2 / 6 

Estado “Reconocidos”

El primer estado de este *alpha* se refiere a los stakeholders como **Reconocidos** (**Recognized**). Esto implica que el equipo de desarrollo tiene identificados a los diferentes grupos de stakeholders, sabe quién representa a cada grupo y ha logrado identificar qué responsabilidad le corresponde, dentro de la iniciativa, a cada uno de ellos.

“Stakeholder groups identified (Grupos de stakeholders identificados)”: El equipo consideró que al momento de iniciado el proceso de desarrollo los grupos de stakeholders se encontraban identificados y separados en tres al momento de la *iteración 0*, considerándose alcanzado este elemento. Los grupos existentes identificados fueron:

- 1) Equipo de desarrollo de la iniciativa
- 2) Organizadores del congreso académico ENIEF 2019
- 3) Asistentes al ENIEF 2019 – usuarios de la aplicación

“Key stakeholder groups represented (Grupos de interesados claves identificados)”: Habiendo previamente identificado los grupos de stakeholders de la iniciativa, se consideró como grupo clave al de *Organizadores del congreso académico ENIEF 2019*. Al momento de la *iteración 0*, era dicho grupo el que tenía poder de decisión respecto de los requerimientos a abordar mediante la solución de software de la iniciativa y era, además, el grupo que se encargaría de proveer la financiación del proyecto.

“Responsibilities defined (Responsabilidades definidas)”: A partir del análisis de la naturaleza de los tres grupos de stakeholders identificados, se consideró al momento de la *iteración 0* que sólo competían responsabilidades a los grupos de 1) Equipo de desarrollo de la iniciativa y 2) Organizadores del congreso académico ENIEF 2019.

Si bien los asistentes/usuarios de la aplicación también son stakeholders, el hecho de que no poseían una responsabilidad definida que afectase al éxito del desarrollo de la iniciativa de software hizo que no fuesen considerados a la hora del análisis de este elemento.

Se definieron, al momento de la iteración 0, las siguientes responsabilidades para los grupos 1 y 2:

- 1) El equipo de desarrollo tendría la responsabilidad de llevar adelante el trabajo para desarrollar la solución de software.
 - 2) El grupo de organizadores del ENIEF 2019 tendría la responsabilidad de aprobar el trabajo realizado a través de las distintas iteraciones de desarrollo, brindando *feedback*, a través de su representante, y encargándose de la provisión del financiamiento económico como contraprestación al trabajo realizado para el desarrollo de la iniciativa.
-

Estado “Representados”

El segundo estado de este *alpha* se refiere a los stakeholders como **Representados (Represented)**. Esto implica que el equipo de desarrollo posee definidos y acordados los mecanismos mediante los cuales los stakeholders estarán involucrados dentro del trabajo que concierne al desarrollo de la iniciativa.

“Responsibilities agreed (Responsabilidades acordadas)”: Definidas para cada grupo de stakeholders las responsabilidades, las mismas fueron también acordadas al momento de la planificación preliminar del presente proyecto. Por ello se consideró a este elemento como alcanzado para la *iteración 0*.

Las responsabilidades sobre las que se generó acuerdo son aquellas definidas en el estado anterior del presente *alpha*.

“Representatives authorized (Representantes autorizados)”: Dentro de los grupos de stakeholders se determinó que sólo uno de los grupos requería contar con una persona autorizada que oficie de representantes: el grupo de organizadores del ENIEF 2019. Desde el comienzo del desarrollo de la iniciativa, el Dr. Pablo Kler fue quien ofició de representante del grupo organizador del ENIEF 2019, con lo cual ya se contaba con un representante autorizado al momento de la *iteración 0*.

“Collaboration approach agreed (Acuerdo en el abordaje para colaborar)”: Habiendo un acuerdo acerca de abordar la colaboración mediante la entrega de incrementos de software planificados y dada a la disponibilidad del representante de los stakeholders para ser consultado vía correo electrónico y videollamadas planificadas según las necesidades del equipo de desarrollo al momento de la planificación, se determinó a este elemento como alcanzado al momento de la *iteración 0*.

“Way of working supported & respected (Way of working adoptado y respetado)”: Sin contar con objeciones respecto de cómo el equipo de desarrollo estableció su proceso y decidió las prácticas a adoptar, comunicando ambos abiertamente frente a los stakeholders, el equipo consideró a este elemento como alcanzado al momento de la *iteración 0*.

Alpha “Equipo”

El equipo de desarrollo consideró que el *alpha* de Equipo, al comienzo de la iniciativa, se encontraba en el estado 3 de 6, que refiere al equipo como “**Colaborando**” (**Collaborating**).

Para el caso de este alpha, en los análisis de logro de cada uno de los elementos dentro de los estados existen menciones múltiples a la posibilidad de cambios en la composición del equipo. Dada la naturaleza de esta iniciativa, la cual está gobernada por las reglas inherentes a lo que dicta la cátedra de Proyecto Final de Carrera, la composición del equipo permaneció inmutable a lo largo de todo el ciclo de desarrollo. Se hace esta aclaración debido a la notable redundancia en los análisis de logro de varios elementos, detalle del cual seguramente el lector podrá tomar nota.

Team	Team	Team
Seeded	Formed	Collaborating
<input checked="" type="checkbox"/> Mission defined <input checked="" type="checkbox"/> Constraints known and defined <input checked="" type="checkbox"/> Growth mechanisms in place <input checked="" type="checkbox"/> Composition defined <input checked="" type="checkbox"/> Responsibilities outlined <input checked="" type="checkbox"/> Required commitment level clear <input checked="" type="checkbox"/> Required competencies identified <input checked="" type="checkbox"/> Size determined <input checked="" type="checkbox"/> Governance rules defined <input checked="" type="checkbox"/> Leadership model selected	<input checked="" type="checkbox"/> Individual responsibilities accepted and aligned to competencies <input checked="" type="checkbox"/> Enough members recruited <input checked="" type="checkbox"/> Roles understood <input checked="" type="checkbox"/> How to work understood <input checked="" type="checkbox"/> Members introduced <input checked="" type="checkbox"/> Members accepting work <input checked="" type="checkbox"/> External collaborators identified <input checked="" type="checkbox"/> Communication mechanisms defined <input checked="" type="checkbox"/> Members commit to team	<input type="checkbox"/> Works as one unit <input checked="" type="checkbox"/> Communication open and honest <input checked="" type="checkbox"/> Focused on mission <input checked="" type="checkbox"/> Members know each other
SEMAIT 1 / 5	SEMAIT 2 / 5	SEMAIT 3 / 5

Estado “Iniciado”

El primer estado de este *alpha* se denomina al equipo como **Iniciado (Seeded)**. Este estado implica que el equipo tiene definido cuál es su misión dentro de la presente iniciativa de software y que conoce cuál es el *know how* que debe generar para cumplir con esta misión.

“Mission defined (Misión definida)”: El equipo consideró que la misión del equipo de desarrollo se encontraba definida para esta etapa, dada la identificación de la oportunidad a abordar y los resultados requeridos por parte del desarrollo de una iniciativa de software que se materializase a partir esa oportunidad.

“Constraints known and defined (Restricciones conocidas y definidas)”: Para esta etapa, el equipo consideró que eran conocidas y estaban definidas las restricciones materiales, técnicas, de tiempo y de esfuerzo que afectarían al desarrollo de la iniciativa.

“Growth mechanisms in place (Mecanismos de agregado de personas el equipo establecido)”: El equipo consideró cumplido este elemento dado que, debido a las restricciones formales para el desarrollo de un Proyecto Final de Carrera, se encontraba ya

establecido que no habría agregado alguno de otras personas a la iniciativa, y que serían Ramiro Olivencia y Federico Albertengo los únicos desarrolladores del equipo a lo largo del ciclo de vida de la iniciativa.

“Composition defined (Composición del equipo definida)”: Como se detalló para el elemento anterior, la composición del equipo ya se encontraba definida para esta etapa y resultaría inmutable hasta finalizada la ejecución de la iniciativa.

“Responsibilities outlined (Responsabilidades delineadas)”: Teniendo ya cimentado y elegido el proceso de software a utilizar, y dada la experiencia de las iniciativas previas encaradas por el equipo, se consideró para esta etapa que las responsabilidades de cada miembro se encontraban delineadas, al menos para una primera iteración del proyecto.

“Required commitment level clear (Aclarado el nivel de compromiso requerido)”: El equipo consideró, para esta etapa, que el conocimiento mutuo de la rutina habitual de los integrantes y cómo esto podía afectar al desarrollo de la iniciativa resultaba suficiente para como para determinar que esto estaba cumplido.

“Required competencies identified (Identificadas las capacidades requeridas)”: Se consideró cumplido este elemento, dado que el mismo estaba conformado por dos personas que contaban con las capacidades suficientes para poder llevar adelante la iniciativa; definición que también contempla la capacidad de poder generar aprendizajes nuevos en el transcurso de la iniciativa para llevarla a buen término en cada incremento realizado.

“Size determined (Tamaño del equipo determinado)”: De forma similar a lo antes explicado, la cantidad de integrantes se mantendría inmutable desde esta etapa hasta finalizado el desarrollo de la iniciativa, por cuestiones formales relacionadas al Proyecto Final de Carrera.

“Governance rules defined (Reglas de autoridad definidas)”: Dado el uso de un proceso de desarrollo de software basado en Kanban y la inexistencia de roles jerárquicos en el equipo de trabajo (heredada del desarrollo de iniciativas anteriores), éste asume que la forma de libre cooperación y vinculación del equipo define una forma de autoridad, en la cual ambos integrantes se encuentran al mismo nivel jerárquico y pueden ambos tener comunicación con los stakeholders externos por igual.

“Leadership model selected (Modelo de liderazgo seleccionado)”: Complementando lo detallado en el elemento anterior, y siendo fiel a los principios que marcan el proceder bajo un proceso ágil, el equipo desde su concepción se consideró autogestivo, en colaboración constante y con una jerarquía horizontal. Por ello, este elemento se consideró cumplido al momento de la *iteración 0*.

Estado “Formado”

El segundo estado de este *alpha* identifica al equipo como **Formado**. Alcanzar este estado implica que el equipo ha sido conformado con la cantidad necesaria de personas como para poder hacer posible el desarrollo de la iniciativa de software.

“Enough members recruited (Suficientes integrantes reclutados)”: Se consideró alcanzado este elemento, al tomar la decisión de emprender la iniciativa con un equipo de dos integrantes, el cual permanecería inmutable hasta el final del desarrollo.

“Roles understood (Roles comprendidos)”: El equipo consideró que cada integrante comprendía su rol dentro del equipo al momento de la *iteración 0*, dada la adopción de una forma de organización horizontal, la cual fue descripta anteriormente en elementos del estado anterior de este *alpha*.

“How to work understood (Comprensión de cómo llevar adelante el trabajo)”: Dada su capacidad de resolución, evidenciada en el desarrollo de iniciativas anteriores, el equipo consideró comprendido cómo llevar adelante el trabajo que la iniciativa de software en cuestión requirió.

“Members introduced (Conocimiento entre integrantes del equipo)”: Dada la existencia de una relación previa de parte de los integrantes del actual equipo, a raíz del desarrollo de iniciativas anteriores en conjunto, se consideró a este elemento como cumplido para la *iteración 0*.

“Individual responsibilities accepted and aligned to competencies (Responsabilidades individuales aceptadas y alineadas a las competencias de los integrantes)”: Tomando como base el mutuo interés de los integrantes del equipo en llevar a buen puerto la iniciativa de software, por razones académicas y en segundo lugar con la posibilidad de generar un beneficio comercial, se consensuaron las responsabilidades que cada integrante debía aceptar, respecto de sus competencias y habilidades, a la hora de generar un equipo de trabajo, cumpliendo así lo prescripto por este elemento para la *iteración 0*.

“Members accepting work (Integrantes aceptando trabajo)”: Dada la naturaleza autogestiva del equipo, el trabajo a aceptar se asignaba consensuando las habilidades de los dos integrantes a través de reuniones periódicas determinadas a tal fin. Manteniéndose este criterio para la iniciativa, en base a la forma de trabajar adoptada en iniciativas anteriores, se decidió considerar a este elemento como cumplido para la *iteración 0*.

“External collaboration identified (Colaboración externa identificada)”: Para el caso de esta iniciativa de software, la colaboración externa con los stakeholders pertenecientes a la organización del ENIEF 2019 se dio a partir de un representante elegido por ellos: el Dr. Pablo Kler, quien ofició desde el primer momento de desarrollo de la presente iniciativa.

“Communication mechanisms defined (Mecanismos de comunicación definidos)”: Las herramientas, canales y ceremonias mediante las cuales se llevaría adelante la comunicación, tanto de manera interna como de manera externa, se encontraba definida para la *iteración 0*. Esto contempla la generación de visibilidad abierta en progreso del proyecto

por parte de todos los stakeholders, siguiendo lo contemplado por el primer principio de Kanban.

“Members commit to team (Integrantes comprometidos con el equipo)”: Para esta etapa, y a partir del trabajo en iniciativas previas, el equipo se encontró desde un primer momento comprometido a colaborar para poder llevar adelante el trabajo que resultase en el desarrollo de la iniciativa de software parte del presente proyecto. Por ello, para la *iteración 0*, se consideró este elemento como cumplido.

Estado “Colaborando”

El tercer estado de este *alpha* refiere a la capacidad del equipo de encontrarse **Colaborando**. Alcanzar este estado implica que quienes integran el equipo ya tiene una relación fluida y, de esta manera, tienen la posibilidad de trabajar de forma sinérgica para llevar adelante el desarrollo de la iniciativa.

Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 0*.

Logrados

“Communication open and honest (Comunicación abierta y honesta)”: Dadas la colaboración del equipo en iniciativas anteriores y las políticas adoptadas como parte del proceso de desarrollo de software basado en Kanban, el cual considera a la visibilidad de información como un principio fundamental, el equipo consideró encontrarse con un marco de comunicación abierta y honesta ya en la *iteración 0* de la presente iniciativa.

“Focused on misión (Con foco en la misión)”: El equipo, desde un primer momento, comprendía en qué consistía su misión en el proyecto y cuál era el conjunto de esfuerzos requeridos para poder llevar el desarrollo del software a buen término. Por ello, se evaluó en la *iteración 0* a este elemento como cumplido.

“Members know each other (Integrantes se conocen unos a los otros)”: Dadas las colaboraciones previas, los dos integrantes del equipo se conocían el uno al otro a la hora de hecha la evaluación para la *iteración 0*. Por tal motivo, se consideró a este elemento como cumplido en esta etapa.

No logrados

“Works as one unit” (Trabaja de manera unificada): El equipo decidió no marcar como alcanzado este elemento en la *iteración 0* debido a que, por causa de las formalidades de esta iniciativa, el equipo podría tener que colaborar, entre sí y con los stakeholders, de una manera diferente respecto a las experiencias anteriores.

Alpha “Trabajo”

El equipo considera que el *alpha* de Trabajo, al comienzo de la iniciativa, se encontraba en el estado 2 de 6, que refiere al trabajo a realizar como “**Preparado**” (**Prepared**).

Work	Work
Initiated	Prepared
<ul style="list-style-type: none"><input checked="" type="checkbox"/> Required result clear<input checked="" type="checkbox"/> Constraints clear<input checked="" type="checkbox"/> Funding stakeholders known<input checked="" type="checkbox"/> Initiator identified<input checked="" type="checkbox"/> Accepting stakeholders known<input checked="" type="checkbox"/> Source of funding clear<input checked="" type="checkbox"/> Priority clear	<ul style="list-style-type: none"><input checked="" type="checkbox"/> Commitment made<input checked="" type="checkbox"/> Cost and effort estimated<input checked="" type="checkbox"/> Resource availability understood<input checked="" type="checkbox"/> Risk exposure understood<input checked="" type="checkbox"/> Acceptance criteria established<input checked="" type="checkbox"/> Sufficiently broken down to start<input checked="" type="checkbox"/> Tasks identified and prioritized<input checked="" type="checkbox"/> Credible plan in place<input checked="" type="checkbox"/> At least one team member ready<input checked="" type="checkbox"/> Integration points defined
SEMAT 1 / 6 [ES]	SEMAT 2 / 6 [ES]

Estado “Iniciado”

El primer estado de este *alpha* refiere al trabajo como **Iniciado**, lo cual implica que lo realizado concerniente a la iniciativa ha sido requerido y que, por ello, hay razones para dar comienzo con el conjunto de acciones tendientes a desarrollar la iniciativa de software.

“Required result clear (Resultados requeridos comprendidos)”: El equipo consideró, para la *iteración 0*, que los resultados requeridos del trabajo que estaba comenzando a realizarse estaban claros. Esto se debió a la similar naturaleza de la presente iniciativa de software con otras que anteriormente el equipo ya había llevado adelante.

“Constraints clear (Restricciones comprendidas)”: Para esta etapa, el equipo tenía en claro cuáles eran las restricciones en la performance para realizar el trabajo de la iniciativa. En este sentido, la no disponibilidad horaria total para ser dedicada al desarrollo del proyecto fue la restricción más notoria respecto a la performance. Debido a que los integrantes del equipo en rutina diaria debían de dedicar tiempo a llevar adelante otras actividades, tanto laborales como académicas, se determinó a esta como la restricción de mayor incidencia en lo concerniente a este *alpha*.

“Funding stakeholders known (Stakeholders que financiarán el Proyecto conocidos)”: Dado lo expuesto anteriormente en el *alpha* de **Oportunidad**, el equipo desde el primer momento identificó cuáles serían los stakeholders que financiarían el proyecto. Por tal motivo, y debido a que no es necesario para la naturaleza de este proyecto el contar obligatoriamente con una financiación, se consideró a este elemento como alcanzado durante la *iteración 0*.

“Initiator identified (Iniciador del trabajo identificado)”: El comenzar con el trabajo a realizar es responsabilidad del equipo, debido a la naturaleza de la iniciativa. Es el equipo quien debe de ponerse en contacto con los stakeholders y comenzar a desarrollar el trabajo, debido a que ya cuenta con una planificación previa desarrollada para saber cómo ejecutar el desarrollo de la iniciativa. Tomando esto en consideración, el equipo determinó que este elemento había sido alcanzado durante la *iteración 0*.

“Accepting stakeholders known (Stakeholders que aceptan los resultados identificados)”: El equipo consideró que, dada la existencia de un representante de los stakeholders colaborando con el equipo ya durante la etapa de la *iteración 0*, sería éste el encargado de hacer las veces de revisor del trabajo a lo largo de las sucesivas iteraciones.

“Source of funding clear (Origen del financiamiento claro)”: Dado que la organización de la cual los stakeholders son parte es la interesada en financiar el trabajo, desde un primer momento se encontraba identificado el origen su financiamiento. Por tanto, se determinó a este elemento como cumplido para la *iteración 0*.

“Priority clear (Prioridades claras)”: Dado que el equipo contó desde un primer momento con una planificación detallada de las tareas a realizar, se consideró que las prioridades del trabajo pendiente se encontraban ya establecidas al momento de la *iteración 0*, logrando cumplirse este elemento.

Estado “Preparado”

El segundo estado de este *alpha* se refiere al trabajo como **Preparado**. Esto implica que se encuentran cumplidas todas las condiciones necesarias para poder dar comienzo al trabajo requerido por la iniciativa.

Este estado fue alcanzado completamente por el equipo en la revisión realizada como parte de la *iteración 0*. El equipo consideró que la documentación generada a partir de la planificación del presente proyecto era un recurso útil para la evaluación de los requisitos de este estado.

“Commitment made (Compromiso realizado)”: Para esta etapa, y debido a las cuestiones formales que hacen al desarrollo del presente PFC, el equipo consideró que el compromiso respecto del trabajo a llevar adelante estuvo realizado desde el momento de la *iteración 0*, considerando entonces este elemento como cumplido.

“Cost and effort estimated (Costo y esfuerzo estimados)”: Dado lo incluido en la planificación del proyecto en lo que respecta a la organización del esfuerzo y la cantidad estimada de esfuerzo que el presente PFC requería para poder ser materializado, el equipo consideró que esta información se encontraba disponible y especificada al momento de la *iteración 0*.

“Resource availability understood (Disponibilidad de recursos comprendida)”: Dada la comprensión respecto de los recursos con los que contaba el equipo al momento de la *iteración 0*, y contando además el equipo con lo especificado en la planificación respecto de

los recursos necesarios para poder llevar adelante la iniciativa, se consideró a este elemento como alcanzado.

“Risk exposure understood (Exposición a riesgos comprendida)”: Existiendo en la planificación un análisis de riesgos, y ya realizado el correspondiente abordaje de estos riesgos mediante un marco que permitiese gestionarlo, el equipo consideró a este elemento alcanzado al momento de realizada la *iteración 0*.

“Acceptance criteria established (Criterio de aceptación establecido)”: Establecidas en la planificación las iteraciones a llevar adelante en el desarrollo del proyecto y detallándose en ellas cuáles serían las historias de usuario que deberían de estar cumplidas al final de cada iteración, se pudo convenir con los *stakeholders* cuáles serían los entregables que deberían de ser desplegados como parte de la interacción entre el equipo y los stakeholders al final de cada iteración. Tomando esto en consideración, el equipo decidió marcar este elemento como alcanzado al momento de la *iteración 0*.

“Sufficiently broken down to start (Trabajo suficientemente desglosado como para empezar)”: Existiendo la especificación del esfuerzo a realizar y contando con las historias de usuario a abordar durante el desarrollo, el equipo consideró que al momento de la *iteración 0* el trabajo estaba lo suficientemente desglosado como para poder dar comienzo a las diversas actividades de desarrollo del PFC. Por tanto, este elemento se determinó como alcanzado.

“Funding in place (Financiamiento en orden)”: Debido a lo expuesto en los estados alcanzados en el *alpha* de Oportunidad y a que se tenía en claro quiénes y en qué condiciones el proyecto contaría con financiación de parte de los *stakeholders*, el equipo decidió determinar a este elemento como alcanzado para el momento de la *iteración 0*.

“At least one team member ready (Al menos un miembro del equipo listo)”: Encontrándose el equipo listo para desarrollar el trabajo a realizar al momento de la *iteración 0*, y contando con todas las herramientas y recursos necesarios para dar comienzo al desarrollo de la iniciativa, se decidió marcar este elemento como alcanzado.

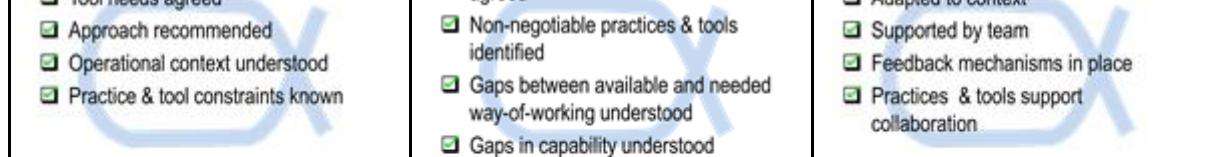
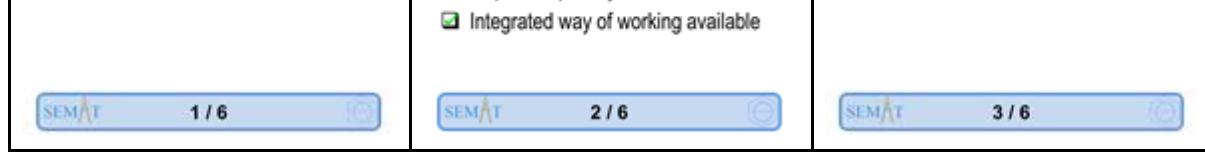
“Credible plan in place (Plan creíble preparado)”: El equipo consideró, al momento de la *iteración 0*, que lo especificado en la planificación del proyecto resultaba en documentación suficiente como para considerar a este elemento como alcanzado.

“Tasks identified and prioritized (Tareas identificadas y priorizadas)”: A partir de lo especificado en la disección de los requerimientos que resultaron en las historias de usuario y contando con la planificación temporal del esfuerzo a invertir en la realización de las tareas, el equipo consideró que al momento de la *iteración 0* este elemento se encontraba alcanzado.

“Integration points defined (Puntos de integración definidos)”: Tomando en consideración la documentación existente en el documento de planificación y la información que se desprende a partir del diagrama de Gantt incluido en ella (el cual posee tiempo dedicado a las pruebas de integración y al despliegue del software) el equipo consideró a este elemento como alcanzado al momento de la *iteración 0*.

Alpha “Way of Working”

El equipo considera que el alpha correspondiente a Way of Working, al comienzo de la iniciativa, se encontraba en el estado 3 de 6, el cual refiere al way of working como “En uso” (“In use”).

Way of Working	Way of Working	Way of Working
Principles Established <ul style="list-style-type: none"><input checked="" type="checkbox"/> Team actively support principles<input checked="" type="checkbox"/> Stakeholders agree with principles<input checked="" type="checkbox"/> Tool needs agreed<input checked="" type="checkbox"/> Approach recommended<input checked="" type="checkbox"/> Operational context understood<input checked="" type="checkbox"/> Practice & tool constraints known  <div style="background-color: #e0f2ff; padding: 5px; display: flex; justify-content: space-between;">SEMAT1 / 6</div>	Foundation Established <ul style="list-style-type: none"><input checked="" type="checkbox"/> Key practices & tools selected<input checked="" type="checkbox"/> Practices needed to start work agreed<input checked="" type="checkbox"/> Non-negotiable practices & tools identified<input checked="" type="checkbox"/> Gaps between available and needed way-of-working understood<input checked="" type="checkbox"/> Gaps in capability understood<input checked="" type="checkbox"/> Integrated way of working available  <div style="background-color: #e0f2ff; padding: 5px; display: flex; justify-content: space-between;">SEMAT2 / 6</div>	In Use <ul style="list-style-type: none"><input checked="" type="checkbox"/> Practices & tools in use<input type="checkbox"/> Regularly inspected<input type="checkbox"/> Adapted to context<input checked="" type="checkbox"/> Supported by team<input checked="" type="checkbox"/> Feedback mechanisms in place<input checked="" type="checkbox"/> Practices & tools support collaboration  <div style="background-color: #e0f2ff; padding: 5px; display: flex; justify-content: space-between;">SEMAT3 / 6</div>

Estado “Principios Establecidos”

El primer estado de este *alpha* se refiere al way of working como con **Principios Establecidos**. Alcanzar este estado implica que el equipo de desarrollo ha consensuado y elegido, tanto internamente como con los stakeholders, los principios bajo los cuales llevará adelante el trabajo requerido para desarrollar la iniciativa de software.

“Team actively support principles (Equipo activamente aplicando los principios)”: En la planificación del presente PFC el equipo eligió aplicar un proceso de desarrollo de software basado en Kanban, el cual requiere de la aplicación y aceptación de un conjunto de principios. Habiéndose comprometido el equipo desde el primer momento a aplicar activamente estos principios como base del way of working, se consideró a este elemento alcanzado al momento de la *iteración 0*.

“Stakeholders agree with principles (Los stakeholders acuerdan con los principios)”: Considerando la aplicación del principio de visualización del trabajo, parte de Kanban, y haciendo partícipes a los stakeholders de la visibilidad del trabajo pendiente y de aquel que a futuro se encontraría en progreso, el equipo consideró alcanzado este elemento al momento de la *iteración 0*.

“Tool needs agreed (Consenso en las herramientas requeridas)”: Teniendo conocimiento de las herramientas de software, hardware y de planificación requeridas para poder llevar adelante una iniciativa de software de estas características (a raíz de iniciativas similares abordadas por el equipo) se consideró al momento de la *iteración 0* a este elemento como alcanzado.

“Approach recommended (Abordaje recomendado)”: Este elemento se define según la existencia o no de un abordaje del *way of working* que se encuentre disponible para consulta por parte del equipo a la hora de generar la implementación del *way of working* propio.

Para el caso de esta iniciativa, y del equipo que la llevaría adelante, se contaba con la experiencia previa de naturaleza académica y profesional por parte de ambos integrantes del equipo. En el aspecto académico, el equipo contaba con experiencia de aprendizaje de herramientas para desarrollo de software con procesos ágiles debido a las asignaturas cursadas y aprobadas como parte de las asignaturas Métodos Ágiles para el Desarrollo de Software e Ingeniería de Software, ambas dentro del plan de estudio de la carrera Ingeniería en Sistemas de Información (Universidad Tecnológica Nacional). Respecto a la experiencia profesional, se encontraba basada en iniciativas de software previas, en las cuales los integrantes aplicaron procesos ágiles para su desarrollo, tanto trabajando en el mismo equipo como en equipos con otras personas en las diversas actividades laborales de cada uno de los integrantes.

“Operational context understood (Contexto operacional comprendido)”: Dada la experiencia previa de trabajo en conjunto de los integrantes del equipo, una forma de operar, basada en el trabajo remoto se encontraba definida al momento de la *iteración 0*. El equipo trabajó en iniciativas anteriores de manera remota, por lo cual planificó el adoptar los mismos criterios para llevar adelante la iniciativa que nuclea al presente PFC. Tomando esto en consideración, el equipo decidió marcar este elemento como alcanzado.

“Practice & tool constraints known (Restricciones de prácticas y herramientas conocidas)”: Conociendo las herramientas que el equipo usaría durante el desarrollo del presente PFC, se tenía información previa acerca de cuáles serían las limitaciones para considerar en lo relacionado a prácticas y herramientas. Teniendo en cuenta que para el desarrollo de la iniciativa el equipo consideró el uso de herramientas disponibles gratuitamente (sin problemas de licenciamiento), y determinándose la utilidad de estas herramientas para el trabajo a realizar, se consideró a este elemento como alcanzado al momento de la *iteración 0*.

Estado “Fundamentos Establecidos”

El segundo estado de este *alpha* se refiere al *way of working* como con sus **Fundamentos Establecidos**. Esto implica que un número mínimo de prácticas centrales y herramientas que serán utilizadas para realizar el trabajo han sido consensuadas, seleccionadas y están listas para ser puestas en uso.

“Key practices & tools selected (Prácticas claves y herramientas seleccionadas)”: Al momento de realizar la planificación ya se encontraban seleccionadas todas las herramientas de software y algunas de las prácticas mínimas necesarias para llevar adelante el trabajo. Por ello, se determinó este elemento como alcanzado para la *iteración 0*.

“Practices needed to start work agreed (Prácticas necesarias para comenzar el trabajo acordadas)”: El conjunto de prácticas que prescribe un proceso de desarrollo basado en Kanban fue considerado como suficiente como para comenzar el trabajo requerido por la

iniciativa. Si bien otras prácticas adicionales a las prescriptas por los principios de Kanban fueron luego utilizadas, al momento de la *iteración 0* las adoptadas por el equipo permitían dar comienzo al trabajo, y por ello se consideró a este elemento como alcanzado.

“Non-negotiable practices & tools identified (Prácticas y herramientas no negociables acordadas)”: El equipo consideró, para este elemento, que las herramientas y prácticas no negociables estaban fuertemente influenciadas por dos aspectos. En primer lugar, por el tipo de iniciativa a llevar adelante, la misma requiere del uso de herramientas para gestión de procesos ágiles y de desarrollo de aplicaciones híbridas. A ello se suma el uso de arquitecturas de tipo cliente-servidor basadas en APIs RESTful, determinando cuáles serían el tipo de prácticas y herramientas necesarias para abordar el proceso y las tecnologías elegidas.

En segundo lugar, existieron restricciones que son impuestas por la naturaleza de los requerimientos académicos por parte del presente PFC, con lo cual hay herramientas (como el Diagrama de Gantt) y prácticas (como la generación de una planificación previa completa por fuera de las iteraciones de desarrollo) que fueron impuestas al equipo.

Teniendo el equipo conocimiento de la naturaleza de estas restricciones y sus consecuencias, el equipo consideró como alcanzado a este elemento.

“Gaps between available and needed way of working understood (Distancia entre el way of working disponible y el necesario comprendido)”: Desde un primer momento, el equipo se comprometió a adoptar un proceso de desarrollo ágil debido a poseer en su haber un conjunto de prácticas heredadas de proyectos anteriores que resultaron también aplicables para el presente proyecto. Por tanto, el equipo consideró que no sólo se comprendía la distancia entre la forma de trabajar actual y la necesaria a ajustar según la planificación y las posteriores disruptpciones y necesidades del proyecto, sino que también el proceso de desarrollo de software elegido permitía la incorporación de nuevas herramientas y prácticas según se precisara. Por ello, se consideró a este elemento como alcanzado al momento de la *iteración 0*.

“Integrated way of working available (Way of working integrado disponible)”: Teniendo el equipo un *way of working* integrado para iniciativas previas, en donde las herramientas y prácticas utilizadas para el desarrollo se encontraban aún disponibles para llevar adelante una nueva iniciativa de software, el equipo determinó a este elemento como alcanzado al momento de la *iteración 0*.

Estado “En Uso”

El tercer estado de este *alpha* se refiere al *way of working* como **En Uso**. Alcanzar este estado implica que al menos algunos de los miembros del equipo de desarrollo están utilizando y adaptando el *way of working* convenido. Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 0*.

Logrados

“Practices & tools in use (Prácticas y herramientas en uso)”: Considerando que el equipo reutilizó las herramientas de desarrollo de software de iniciativas anteriores y acordó las prácticas iniciales que darían soporte a la metodología de desarrollo, con la posibilidad de incorporar prácticas nuevas mientras el trabajo se encontraba en curso, se determinó a este elemento como alcanzado para el momento de la *iteración 0*.

“Supported by team (Soportado por el equipo)”: Siendo el mismo equipo quien decidió y definió en un primer momento el *way of working*, en base a la experiencia de iniciativas previas, se asume que éste se encuentra soportado por el equipo. Es por ello que el equipo, además, se sintió a gusto con el *way of working* adoptado y fue capaz, posteriormente a la *iteración 0*, de evaluar las necesidades de modificarlo según avanzó el proyecto.

“Feedback mechanisms in place (Mecanismos de realimentación en orden)”: Habiendo el equipo decidido adoptar una metodología de desarrollo basada en una implementación de Kanban, se desprende la existencia de mecanismos de *feedback* aplicables al desarrollo del presente proyecto que son prescriptos por la metodología. Tomando en cuenta los principios de Kanban, los cuales consideran la revisión continua del proceso como parte esencial del framework, más las iteraciones y retrospectivas que el equipo fijó para el desarrollo del presente proyecto, se consideró a este elemento como alcanzado al momento de la *iteración 0*.

“Practices & tools support collaboration (Prácticas y herramientas dan soporte a la colaboración)”: Tomando en consideración la experiencia previa con prácticas y herramientas que fueron elegidas para brindar soporte al desarrollo de la presente iniciativa, el equipo consideró que al momento de la *iteración 0* tanto prácticas como herramientas cumplían con la característica de dar soporte a la colaboración de los integrantes del equipo y de generar un trabajo más eficiente y efectivo.

No logrados

“Regularly inspected (Regularmente inspeccionado)”: Dado que la línea temporal al momento de la *iteración 0* no permite una inspección del *way of working* respecto del trabajo en realización (el cual aún no había sido comenzado para este momento por fuera de la planificación preliminar) el equipo consideró a este elemento como no alcanzado.

“Adapted to context (Adaptado al contexto)”: Dado que el trabajo orientado al diseño y la programación del proyecto no había sido comenzado al momento de la *iteración 0*, y con ello no se asumía aún una completa comprensión acerca del contexto dentro del cual se llevaría adelante el trabajo y la gestión de este, el equipo asumió que no estaban dadas las condiciones para asumir que el *way of working* cumplía con la característica de encontrarse adaptado al contexto.

Iteración 1

La iteración 1 se llevó adelante entre los días 27 de mayo y 21 de junio de 2019.

Objetivos de la iteración

Acorde a lo detallado en la planificación, la primera iteración del proyecto tuvo como objetivo:

- Desarrollar parcialmente la infraestructura necesaria para proveer a la aplicación de un servidor web y una base de datos.
 - Determinar completamente la configuración inicial del servidor web.
 - Crear completamente la estructura inicial de la base de datos relacional.
 - Desarrollar parcialmente la colección de *endpoints* de las APIs RESTful del servidor web, entregando como completados en la iteración 1 los correspondientes a los módulos de "Actividades y Cronograma".
- Desarrollar un incremento para el módulo "Actividades y Cronogramas" de la aplicación.
 - Desarrollo de la vista "Próxima Actividad" dentro del módulo.
 - Desarrollo del servicio "Próxima Actividad" como parte del módulo.
 - Desarrollo de la lógica de filtrado y selección de actividades.
 - Desarrollo de la funcionalidad "Actividades Favoritas".
- Gestionar el proceso de desarrollo de software basado en Kanban.
- Llevar adelante reuniones con el representante de los *stakeholders*.
- Realizar una demostración en vivo al representante de los *stakeholders* del primer entregable funcional de software.

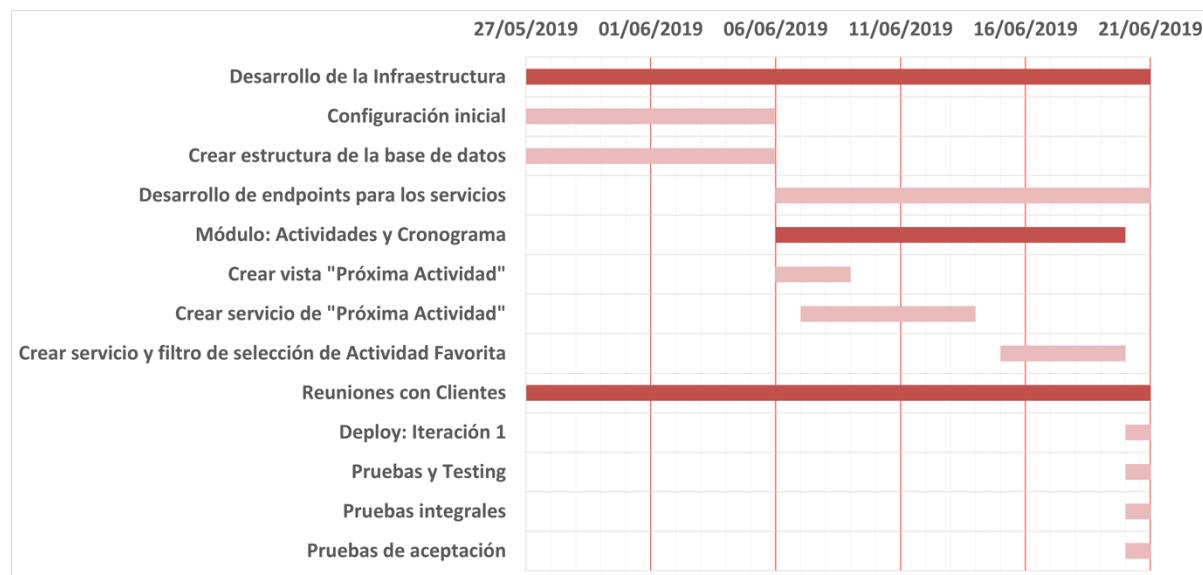


Figura 27 – Recorte de la iteración 1 del diagrama de Gantt del plan de proyecto³¹.

³¹Se puede encontrar en formato digital en <https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Diagrama%20de%20Gantt/Gantt%20Iteracion%201.png>.

Revisión de los estados alcanzados

<h3>Requirements</h3> <p>Bounded</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Development stakeholders identified <input checked="" type="checkbox"/> System purpose agreed <input checked="" type="checkbox"/> System success clear <input checked="" type="checkbox"/> Shared solution understanding exists <input type="checkbox"/> Requirement's format agreed <input checked="" type="checkbox"/> Requirements management in place <input checked="" type="checkbox"/> Prioritization scheme clear <input type="checkbox"/> Constraints identified & considered <input type="checkbox"/> Assumptions clear <p>SEMAT 2 / 6</p>	<h3>Software System</h3> <p>Demonstrable</p> <ul style="list-style-type: none"> <input type="checkbox"/> Key architectural characteristics demonstrated <input checked="" type="checkbox"/> System exercised & performance measured <input type="checkbox"/> Critical HW configurations demonstrated <input type="checkbox"/> Critical interfaces demonstrated <input type="checkbox"/> Integration with environment demonstrated <input checked="" type="checkbox"/> Architecture accepted as fit-for-purpose <p>SEMAT 2 / 6</p>	
<h3>Opportunity</h3> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <p>SEMAT 4 / 6</p>	<h3>Stakeholders</h3> <p>Represented</p> <ul style="list-style-type: none"> <input type="checkbox"/> Responsibilities agreed <input checked="" type="checkbox"/> Representatives authorized <input checked="" type="checkbox"/> Collaboration approach agreed <input checked="" type="checkbox"/> Way of working supported & respected <p>SEMAT 2 / 6</p>	
<h3>Team</h3> <p>Collaborating</p> <ul style="list-style-type: none"> <input type="checkbox"/> Works as one unit <input checked="" type="checkbox"/> Communication open and honest <input checked="" type="checkbox"/> Focused on mission <input checked="" type="checkbox"/> Members know each other <p>SEMAT 3 / 5</p>	<h3>Work</h3> <p>Prepared</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Commitment made <input checked="" type="checkbox"/> Cost and effort estimated <input checked="" type="checkbox"/> Resource availability understood <input checked="" type="checkbox"/> Risk exposure understood <input checked="" type="checkbox"/> Acceptance criteria established <input checked="" type="checkbox"/> Sufficiently broken down to start <input checked="" type="checkbox"/> Tasks identified and prioritized <input checked="" type="checkbox"/> Credible plan in place <input checked="" type="checkbox"/> At least one team member ready <input checked="" type="checkbox"/> Integration points defined <p>SEMAT 2 / 6</p>	<h3>Way of Working</h3> <p>In Use</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Practices & tools in use <input type="checkbox"/> Regularly inspected <input type="checkbox"/> Adapted to context <input checked="" type="checkbox"/> Supported by team <input checked="" type="checkbox"/> Feedback mechanisms in place <input checked="" type="checkbox"/> Practices & tools support collaboration <p>SEMAT 3 / 6</p>

Planificación de la iteración

Llevar adelante la primera iteración implicó la primera ronda de asignación y gestión del trabajo relacionado a las dos áreas arquitectónicamente significativas del proyecto, definidas por 1) el servidor web con el cual se comunica la aplicación móvil y 2) el incremento de software de la aplicación móvil en sí misma.

En base a ello, el equipo dividió el trabajo relacionado al desarrollo del servidor web y el desarrollo relacionado al incremento de la aplicación web, dividiendo el trabajo relacionado a las historias de usuario concernientes a esta iteración en tareas de desarrollo. Posteriormente a la determinación y detalle del trabajo a realizar las tareas se asignaron a los miembros del equipo.

Dada la obvia vinculación entre los dos módulos involucrados en el desarrollo, durante esta primera iteración, y luego durante todo el desarrollo relativo a la ejecución del proyecto, fue esencial la comunicación constante del equipo y la prueba de los sucesivos incrementos producto de la conclusión de cada tarea de desarrollo. Para lograr estos objetivos el equipo llevó adelante pruebas de integración para validar el funcionamiento de las interfaces entre aplicación y servidor.

Además de lo relativo a la programación de los módulos, el equipo planificó el comenzar a poner a punto las herramientas de soporte, lo cual le permitiría llevar adelante la gestión de las tareas relevantes a las actividades relacionadas al desarrollo de los módulos mencionados.

Alpha "Requerimientos"

Requirements	Requirements	Requirements
Bounded	Coherent	Acceptable
<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Development stakeholders identified <input checked="" type="checkbox"/> System purpose agreed <input checked="" type="checkbox"/> System success clear <input checked="" type="checkbox"/> Shared solution understanding exists <input checked="" type="checkbox"/> Requirement's format agreed <input checked="" type="checkbox"/> Requirements management in place <input checked="" type="checkbox"/> Prioritization scheme clear <input checked="" type="checkbox"/> Constraints identified & considered <input checked="" type="checkbox"/> Assumptions clear 	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Requirements shared <input checked="" type="checkbox"/> Requirements' origin clear <input checked="" type="checkbox"/> Rationale clear <input checked="" type="checkbox"/> Conflicts addressed <input checked="" type="checkbox"/> Essential characteristics clear <input checked="" type="checkbox"/> Key usage scenarios explained <input checked="" type="checkbox"/> Priorities clear <input checked="" type="checkbox"/> Impact understood <input checked="" type="checkbox"/> Team knows & agrees on what to deliver 	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable
SEM/T 2 / 6	SEM/T 3 / 6	SEM/T 4 / 6

Como resultado de la iteración 1, el *alpha* Requerimientos reflejó un avance en la progresión de estados, transicionando desde el estado “Acotados”, segundo en la escala de seis, hacia el estado “Aceptables”, cuyo logro abordado parcialmente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

Estado “Acotados”

El segundo estado de este *alpha*, Requerimientos “**Acotados**”, refiere a la comprensión del propósito y el contexto dentro del cual el sistema debe de operar. A partir de la iteración anterior, se mencionan los nuevos elementos logrados de este estado, considerándose que este estado ha sido alcanzado completamente.

“Requirement's format agreed” (Consenso en el formato de los requerimientos): Durante la *iteración 1* se logró una forma consensuada respecto a cómo tratar los nuevos requerimientos que surgieran durante la actual y posteriores iteraciones del desarrollo, las cuales resultaban en modificaciones o refinamientos respecto de las actividades y tareas planteadas en la planificación original.

Para llevar registro de los nuevos requerimientos o las modificaciones en los existentes el equipo y los stakeholders convinieron en utilizar las minutos obtenidas a partir de las reuniones con los clientes y utilizar la herramienta Trello para plasmar y refinar las modificaciones y refinamientos surgidos a partir de estas reuniones, siguiendo el proceso ágil basado en Kanban detallado en la sección “Proceso de Desarrollo”.

“Assumptions are clearly stated” (Suposiciones claramente declaradas): A través de las reuniones realizadas, una especificación clara y actualizada del producto en desarrollo y un backlog compartido refinado, se considera que las suposiciones respecto de todo lo que se requiere que el sistema haga fueron establecidas.

“Constraints identified & considered” (Limitaciones identificadas y consideradas): El equipo, tras hacer una revisión del análisis de riesgos a detalle, basada en la planificación del presente Proyecto Final de Carrera, determinó que este elemento se puede considerar como resuelto.

Estado “Coherentes”

El tercer estado de este *alpha*, requerimientos “**Coherentes**”, refiere a que los requerimientos proveen una descripción consistente de las características esenciales del sistema. Se mencionan los nuevos elementos alcanzados para este estado en la Iteración 1.

“Requirements shared” (Requerimientos compartidos): El equipo considera que los requerimientos han sido capturados y compartidos con los stakeholders mediante la herramienta Trello, oficiando ésta como radiador de información hacia el equipo de desarrollo y los stakeholders. Por ello, se determinó a este elemento como alcanzado.

“Requirements' origin clear” (Origen de los requerimientos claro): A partir de las reuniones se identificaron las necesidades a abordar que dan origen a cada requerimiento planteado, por lo que puede considerarse que este elemento ha sido cumplido.

“Rationale clear” (Razón fundamental clara): Hace referencia a la razón fundamental, esencial, que existe detrás de los requerimientos planteados. Luego de evaluaciones sucesivas entre el equipo y los stakeholders, se consideró a este elemento como alcanzado.

“Conflicts addressed” (Conflictos abordados): Debido a la capacidad de organización de las tareas y el trabajo que brindan las herramientas, las prácticas y el proceso seleccionados, se pudieron establecer cuáles eran los requerimientos que tenían conflictos entre sí. De esta forma, fue posible luego organizar las tareas de forma que se puedan resolver de la mejor forma posible.

“Essential characteristics clear” (Características esenciales claras): Se considera que las características esenciales se han definido claramente, habiendo los stakeholders evaluado las características ofrecidas por los incrementos anteriores y teniendo un conocimiento claro respecto de las características que el incremento a desarrollar proveería para el producto. Por eso se considera que este elemento ha sido cumplido.

“Key usage scenarios explained” (Escenarios de uso principales explicados): Los escenarios de uso han sido compartidos y explicados por el equipo a stakeholders llegando a un entendimiento mutuo sobre los mismos. Para ello, el equipo tomó como soporte la evidencia del éxito de los incrementos previos en contextos similares a los requeridos por presente proyecto y documentación para el presente incremento.

“Priorities clear” (Prioridades claras): Las prioridades sobre los requerimientos y tareas a realizar están establecidas y aceptadas por el equipo y los stakeholders.

“Impact understood” (Impacto entendido): Esto refiere al entendimiento sobre el impacto que produce implementar los requerimientos definidos. Esto se considera cumplido debido a que, gracias a lo evaluado en la planificación del presente proyecto, ya se cuenta con experiencia sobre el impacto que pueden producir ciertos requerimientos y se ha analizado el impacto y el riesgo de los nuevos.

“Team knows & agrees on what to deliver” (El equipo conoce y acepta lo que se va a entregar): Este elemento hace alusión a que el equipo conoce y acepta lo que los stakeholders esperan que se entregue a partir de los requerimientos planteados. Este elemento se puede considerar alcanzado.

Estado “Aceptados”

El cuarto estado de este *alpha*, Requerimientos “**Aceptados**”, refiere a que los requerimientos describen un sistema que es aceptable para los stakeholders. A continuación, se exponen los requisitos alcanzados y no alcanzados de este estado.

Logrados

“Acceptable solution described” (Solución descrita aceptable): Se refiere a que los stakeholders aceptaron que los requerimientos describen una solución aceptable por ellos. Esto se consideró alcanzado, dada la disponibilidad de los requerimientos refinados y documentados en alto nivel.

“Value to be realized clear” (Valor a realizar claro): Los stakeholders son conscientes del valor que provee el hecho de implementar los requerimientos.

“Clear how opportunity addressed” (Forma de abordar la oportunidad clara): Tanto el equipo como los stakeholders tienen claro cómo los requerimientos permitirán abordar la oportunidad que refiere al proyecto, detallando en alto nivel cómo cada requerimiento será a su vez abordado por las distintas características implementadas en el sistema de software..

“Testable” (Verifiable): Este elemento hace referencia a que existe un proceso finito y no costoso para demostrar que el sistema cumple con un requerimiento. Para esto, como se desarrolló en la sección correspondiente, se utilizaron pruebas unitarias y de aceptación. Por ello, este elemento se determina como alcanzado.

No logrados

“Change under control” (Cambios bajo control): Este elemento está relacionado con que la tasa de cambio en relación con los requerimientos aceptados es relativamente baja y esa tasa se mantiene en un umbral relativamente estable. Debido a que el equipo consideró que esta etapa era demasiado temprana como para asegurar el cumplimiento de este elemento, se decidió no marcarlo como logrado.

Software System

Demonstrable

- Key architectural characteristics demonstrated
- System exercised & performance measured
- Critical HW configurations demonstrated
- Critical interfaces demonstrated
- Integration with environment demonstrated
- Architecture accepted as fit-for-purpose

SEMANT 2 / 6

Como resultado de la iteración 1, el *alpha* Sistema de Software reflejó un avance en la progresión de elementos, manteniéndose dentro del estado “Demostrable”, segundo en la escala de seis.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las progresiones en el alcance de los elementos componentes del estado alcanzado.

Estado “Demostrable”

El segundo estado de este *alpha* se refiere al sistema de software como **Demostrable**. Esto implica que existe una versión del sistema de software ejecutable, la cual permite demostrar que la arquitectura es adecuada para el propósito del sistema y que, además, soporta la posibilidad de poder realizar pruebas para validarla.

“Key architectural characteristics demonstrated (Características claves de la arquitectura demostradas)”: Producto del progreso en el desarrollo del proyecto alcanzado en la iteración 1 se vuelve posible demostrar la funcionalidad y las características claves de la arquitectura seleccionada para el proyecto.

Finalizada la iteración 1 el equipo poseyó suficientes características desarrolladas en el servidor web y la aplicación móvil como para poder llevar adelante una demostración de funcionamiento de ambas partes del sistema, las cuales reflejan la arquitectura seleccionada.

“Critical hardware configurations demonstrated (Configuración crítica de hardware demostrada)”: Al tener un entregable desplegado para la iteración 1, resulta posible la demostración de la configuración crítica del hardware necesaria.

“Critical interfaces demonstrated (Interfaces críticas demostradas)”: Al contar con un software utilizable referido a los nuevos requerimientos con interfaces demostrables referidas

al incremento que representa a la iniciativa del PFC, el equipo decidió considerar a este elemento como alcanzado.

“Integration with environment demonstrated (Integración con el ambiente demostrada)”: Se ha evaluado el software desplegado para la presente iniciativa, para poder demostrar la integración del software con el ambiente en el cual se ejecutaría o uno similar.

Alpha “Oportunidad”

Opportunity

Viable

- Solution outlined
- Solution possible within constraints
- Risks acceptable & manageable
- Solution profitable
- Reasons to develop solution understood
- Pursuit viable

SEMANT4 / 6(2)

Como resultado de la iteración 1, el *alpha* Oportunidad no reflejó ningún avance en la progresión de estados, quedando en el estado “Viable”, cuarto en la escala de seis, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 0.

Alpha “Stakeholders”

Stakeholders

Involved

- Representatives assist the team
- Timely feedback and decisions provided
- Changes promptly communicated



SEMAF3 / 6

Como resultado de la iteración 1, el *alpha Stakeholders* reflejó un avance en la progresión de elementos, pasando al del estado “Involucrados”, tercero en la escala de seis.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las progresiones en el alcance de los elementos componentes del estado alcanzado.

Estado “Involucrados”

El tercer estado de este *Alpha* se refiere a los stakeholders como **Involucrados**. Esto significa que los representantes de los stakeholders se encuentran ya activamente involucrados y colaborando en el desarrollo del sistema, posibilitando la ejecución de las tareas y el trabajo por parte del equipo.

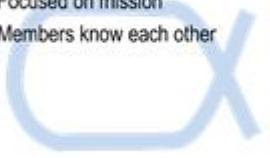
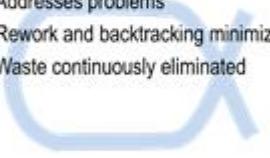
Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 1*.

“Representatives assist the team” (Representantes asisten al equipo): El Dr. Pablo Kler, representante asignado de los stakeholders, estuvo involucrado en el proceso asistiendo al equipo en las dudas o peticiones de información que surgían. Por esto mismo se puede considerar que el elemento ha sido cumplido.

“Timely feedback and decisiones provided” (Periódicamente se proveen decisiones y feedback): Debido a las reuniones periódicas con el representante, se proveían constantemente decisiones y feedback sobre el sistema.

“Changes promptly communicated” (Cambios comunicados con prontitud): En esta iteración ya se pudo alcanzar un nivel de comunicación eficiente con el representante, por lo que los cambios eran comunicados al equipo de desarrollo con rapidez. Por ello este elemento se consideró como alcanzado.

Alpha “Equipo”

Team	Team
Collaborating  <ul style="list-style-type: none"><input checked="" type="checkbox"/> Works as one unit<input checked="" type="checkbox"/> Communication open and honest<input checked="" type="checkbox"/> Focused on mission<input checked="" type="checkbox"/> Members know each other	Performing  <ul style="list-style-type: none"><input checked="" type="checkbox"/> Consistently meeting commitments<input checked="" type="checkbox"/> Continuously adapting to change<input checked="" type="checkbox"/> Addresses problems<input type="checkbox"/> Rework and backtracking minimized<input type="checkbox"/> Waste continuously eliminated
SEMAT 3 / 5 	SEMAT 4 / 5 

Como resultado de la iteración 1, el *alpha* Equipo reflejó un avance en la progresión de estados, transicionando desde el estado “Colaborando”, tercero en la escala de cinco, hacia el estado “Desempeñándose”, cuyo logro fue abordado parcialmente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

Estado “Colaborando”

El tercer estado de este *alpha* refiere a la capacidad del equipo de encontrarse **Colaborando**. Alcanzar este estado implica que quienes integran el equipo ya tiene una relación fluida y, de esta manera, tienen la posibilidad de trabajar de forma sinérgica para llevar adelante el desarrollo de la iniciativa. A partir de la iteración anterior, se mencionan los nuevos requisitos logrados de este estado, considerándose que este estado ha sido alcanzado.

“Works as one unit” (Trabajar como una unidad): A diferencia de la iteración anterior, este elemento se ha marcado como alcanzado debido a que ya se ha podido definir la forma en la que el equipo colabora entre sí y con los *stakeholders* como parte del trabajo concerniente al proyecto.

Estado “Desempeñándose”

El cuarto estado de este *alpha* refiere a la capacidad del equipo de encontrarse **Desempeñándose**. Alcanzar este estado implica que quienes integran el equipo ya están trabajando de forma efectiva y eficiente. A partir de la iteración anterior, se mencionan los nuevos requisitos logrados y no logrados de este estado, considerándose que este estado ha sido alcanzado.

Logrados

“Consistently meeting commitments” (Cumplimiento constante de los compromisos):

Se puede considerar cumplido ya que el equipo, para esta etapa, se encontró cumpliendo constantemente con los objetivos planteados según la planificación y sobre las peticiones de los *stakeholders*.

“Continuously adapting to change” (adaptándose continuamente al cambio): Por la naturaleza ágil del proyecto, se puede considerar cumplido este elemento debido a que se realizan constantemente reuniones y *feedback* de los clientes para realizar cambios y mejoras.

“Adresses problems” (Aborda problemas): Esto refiere a que el equipo se demostró capaz, ya en esta etapa, de generar una manera ordenada y eficaz de identificar y resolver los diversos problemas producto de la ejecución del desarrollo del proyecto. Por ello, este elemento se consideró como logrado.

No logrados

“Rework and backtracking minimized” (Retrabajo y retroceso minimizado): Se relaciona con que el progreso efectivo se está alcanzando con el retrabajo y retroceso mínimo. Este elemento se consideró no cumplido completamente debido a que el equipo se encontraba en una etapa temprana del proyecto, por lo que todavía no se tiene una seguridad completa acerca de mantener en un mínimo el retrabajo.

“Waste continuously eliminated” (Desperdicio continuamente eliminado): Esto refiere a que el trabajo desperdiciado, así como la probabilidad de trabajo desperdiciado se reducen continuamente mientras el proyecto avanza. Esto se considera que no se ha cumplido parcialmente debido a que el equipo se encuentra en una etapa temprana del proyecto, por lo cual el equipo aún no tiene la suficiente seguridad como para poder marcar como alcanzado este elemento.

Alpha “Trabajo”



Como resultado de la iteración 1, el *alpha* Trabajo reflejó un avance en la progresión de elementos, pasando al del estado “Comenzado”, tercero en la escala de seis.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las progresiones en el alcance de los elementos componentes del estado alcanzado.

Estado “Comenzado”

El tercer estado de este *alpha*, “**Comenzado**”, refiere a que el trabajo ha comenzado. A continuación, se exponen los elementos alcanzados de este estado a partir de la iteración anterior.

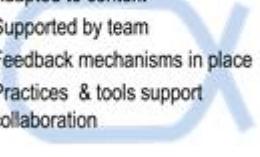
“Development started” (Desarrollo comenzado): Significa que el desarrollo del proyecto ha comenzado. Al momento de finalizada esta iteración ya se encontraban desarrolladas funcionalidades pertenecientes a la aplicación móvil y funcionalidades correspondientes al servidor web.

“Progress monitored” (Progreso monitoreado): Para esta etapa ya el progreso del equipo se encontraba constantemente monitoreado por los *stakeholders* por medio de las reuniones periódicas y podía ser controlado mediante la observación del progreso visible en el tablero de Trello, lográndose así lo requerido para este elemento.

“Definition of done in place” (Definición de hecho establecida): Habiendo el equipo definido en una instancia anterior una primera definición de hecho soportada por las prácticas seleccionadas para el proceso, se determinó a este elemento como alcanzado.

“Task being progressed” (Tareas siendo avanzadas): Los miembros del equipo se asignan las tareas y trabajan sobre ellas de forma organizada y eficiente, evidenciándose un progreso claro en el trabajo asignado para el desarrollo de las funcionalidades.

Alpha “Way of Working”

Way of Working	Way of Working
In Use <ul style="list-style-type: none"><input checked="" type="checkbox"/> Practices & tools in use<input checked="" type="checkbox"/> Regularly inspected<input checked="" type="checkbox"/> Adapted to context<input checked="" type="checkbox"/> Supported by team<input checked="" type="checkbox"/> Feedback mechanisms in place<input checked="" type="checkbox"/> Practices & tools support collaboration	In Place <ul style="list-style-type: none"><input checked="" type="checkbox"/> Used by whole team<input checked="" type="checkbox"/> Accessible to whole team<input checked="" type="checkbox"/> Inspected and adapted by whole team
 SEMAT 3 / 6	 SEMAT 4 / 6

Como resultado de la iteración 1, el *alpha Way of Working* reflejó un avance en la progresión de estados, transicionando desde el estado “En Uso”, tercero en la escala de seis, hacia el estado “En Orden”, cuyo logro fue abordado parcialmente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

Estado “En Uso”

El tercer estado de este *alpha* se refiere al *way of working* como **En Uso (In Use)**. Alcanzar este estado implica que al menos algunos de los miembros del equipo de desarrollo están utilizando y adaptando el *way of working* convenido. Este estado fue completamente alcanzado por el equipo en la revisión realizada como parte de la *iteración 1*.

“Regularly inspected (Regularmente inspeccionado)”: Dado que el trabajo ya ha comenzado, se realizan inspecciones del *way of working* regularmente, sea durante la ejecución habitual de las tareas o en las ceremonias de retrospectiva. Por ello en esta etapa se consideró a este elemento como alcanzado.

“Adapted to context (Adaptado al contexto)”: Dado que el trabajo orientado al diseño y la programación del proyecto ya había comenzado en la iteración 1, y ya se poseía una comprensión acerca del contexto dentro del cual se llevó adelante el trabajo y la gestión de este, se comenzó a adaptar el *way of working* al contexto, considerando a este elemento como alcanzado.

Estado “En Orden”

El cuarto estado de este *alpha* se refiere al *way of working* como **En Orden (In Place)**. Alcanzar este estado implica que todos los miembros del equipo de desarrollo están utilizando

y adaptando el *way of working* convenido. Este estado fue completamente alcanzado por el equipo en la revisión realizada como parte de la *iteración 1*.

“Used by whole team” (Usado por todo el equipo): Este elemento hace alusión a que todo el equipo está utilizando el *way of working* para realizar el trabajo. Se consideró a este elemento como alcanzado para esta etapa del proyecto.

“Accesible to whole team” (Accesible para todo el equipo): Se refiere a que todos los miembros del equipo tienen acceso a las prácticas y a las herramientas requeridas para hacer el trabajo. Se consideró a este elemento alcanzado, fundamentándose el equipo en las políticas de desarrollo adoptadas, basadas en la puesta en práctica del primer principio de Kanban.

“Inspected and adapted by whole team” (Inspeccionado y adaptado por todo el equipo): Desde el comienzo de la ejecución del proyecto todo el equipo se encargó de la inspección y adaptación del *way of working*, con lo cual se consideró a este elemento como alcanzado.

Retrospectiva

Se describe en esta sección una breve explicación de la transición de estados y de elementos alcanzados como parte de la iteración, producto de la evaluación del equipo del trabajo realizado al final de la iteración 1.

Alpha “Requerimientos”

La iteración comenzó con un logro parcial del estado de requerimientos **“Abordados” (Bounded – 2/6)** y finalizó con un logro parcial del estado de requerimientos **“Aceptables” (Acceptable – 4/6)**, visualizándose un marcado progreso en los estados de este *Alpha*. Para lograr esta marcada progresión se destaca el alto grado de colaboración entre el equipo de desarrollo y el representante de los stakeholders.

Alpha “Sistema de Software”

La iteración comenzó con un logro parcial del estado de sistema de software **“Demostrable” (Demonstrable – 2/6)** y finalizó con el logro completo de todos los elementos de este estado, marcando progreso hacia el objetivo de permitir que, con parte de su funcionalidad, el sistema de software pueda ser presentado al final de la iteración al representante de los stakeholders.

Alpha “Oportunidad”

La iteración comenzó y terminó con un logro parcial del estado de oportunidad **“Viable” (Viable – 4/6)**, sin modificación alguna en los estados alcanzados. Luego de marcar como logrados varios estados en la evaluación de la *iteración 0* inicial, este *alpha* no marcó progreso alguno durante la primera iteración.

Alpha “Stakeholders”

La iteración comenzó con un logro completo del estado de los stakeholders como **“Representados” (Represented – 2/6)** dentro de la iniciativa de software y finalizó con un logro completo del estado **“Involucrados” (Involved – 3/6)**. La progresión de estados se justifica en la notable colaboración desarrollada entre el equipo de desarrollo y el representante de los stakeholders a lo largo de la iteración, demostrándose la colaboración útil para poder llevar adelante el trabajo y clarificar requerimientos.

Alpha “Equipo”

La iteración comenzó con un logro completo del estado del equipo como **“Colaborando” (Collaborating – 3/5)** y finalizó con un logro parcial del estado del equipo como **“Desempeñándose” (Performing – 4/5)**. La progresión obedeció a la adaptación del equipo a trabajar con el objetivo en mente de trabajar como una única unidad, comenzar con el cumplimiento de los compromisos detallados en la planificación y comenzar con la generación de entregables de valor producto del trabajo.

Alpha “Trabajo”

La iteración comenzó con un logro parcial del estado del trabajo como **“Preparado” (Prepared – 2/6)** y finalizó con un logro completo del estado del trabajo como **“Comenzado” (Started – 3/6)**. La progresión de estados obedeció, al comienzo del proyecto, a contar con un consenso establecido respecto de la *definición de hecho* para las tareas e historias de

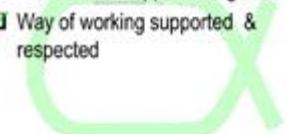
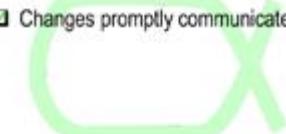
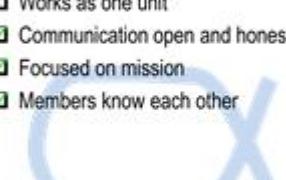
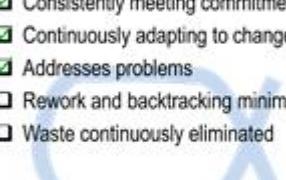
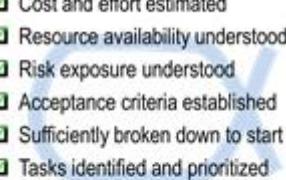
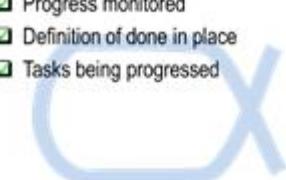
usuario del proyecto y la puesta en práctica del monitoreo constante del progreso realizado por el equipo.

Alpha “Way of Working”

La iteración comenzó con un logro parcial del estado del way of working como “**En Uso**” (**In Use – 3/6**) y finalizó con un logro completo del estado del way of working como “**En Orden**” (**In Place – 4/6**). La progresión de estados se justifica en que el comienzo del trabajo en la iteración 1 permitió la inspección y adaptación del way of working, permitiendo esto a su vez hacerlo visible y accesible al equipo y al representante de los stakeholders.

Resumen de cambios de estado durante la iteración

Alpha	Estado Inicial	Estado Final
Requerimientos	<p>Requirements</p> <p>Bounded</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Development stakeholders identified <input checked="" type="checkbox"/> System purpose agreed <input checked="" type="checkbox"/> System success clear <input checked="" type="checkbox"/> Shared solution understanding exists <input type="checkbox"/> Requirement's format agreed <input checked="" type="checkbox"/> Requirements management in place <input checked="" type="checkbox"/> Prioritization scheme clear <input type="checkbox"/> Constraints identified & considered <input type="checkbox"/> Assumptions clear <p>SEM/T 2 / 6</p>	<p>Requirements</p> <p>Acceptable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable <p>SEM/T 4 / 6</p>
Sistema de Software	<p>Software System</p> <p>Demonstrable</p> <ul style="list-style-type: none"> <input type="checkbox"/> Key architectural characteristics demonstrated <input checked="" type="checkbox"/> System exercised & performance measured <input type="checkbox"/> Critical HW configurations demonstrated <input type="checkbox"/> Critical interfaces demonstrated <input type="checkbox"/> Integration with environment demonstrated <input checked="" type="checkbox"/> Architecture accepted as fit-for-purpose <p>SEM/T 2 / 6</p>	<p>Software System</p> <p>Demonstrable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Key architectural characteristics demonstrated <input checked="" type="checkbox"/> System exercised & performance measured <input checked="" type="checkbox"/> Critical HW configurations demonstrated <input checked="" type="checkbox"/> Critical interfaces demonstrated <input checked="" type="checkbox"/> Integration with environment demonstrated <input checked="" type="checkbox"/> Architecture accepted as fit-for-purpose <p>SEM/T 2 / 6</p>
Oportunidad	<p>Opportunity</p> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <p>SEM/T 4 / 6</p>	<p>Opportunity</p> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <p>SEM/T 4 / 6</p>

	<p>Stakeholders</p> <p>Stakeholders</p> <p>Represented</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Responsibilities agreed <input checked="" type="checkbox"/> Representatives authorized <input checked="" type="checkbox"/> Collaboration approach agreed <input checked="" type="checkbox"/> Way of working supported & respected  <p>SEM/T 2 / 6</p>	<p>Stakeholders</p> <p>Involved</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Representatives assist the team <input checked="" type="checkbox"/> Timely feedback and decisions provided <input checked="" type="checkbox"/> Changes promptly communicated  <p>SEM/T 3 / 6</p>
Equipo	<p>Team</p> <p>Collaborating</p> <ul style="list-style-type: none"> <input type="checkbox"/> Works as one unit <input checked="" type="checkbox"/> Communication open and honest <input checked="" type="checkbox"/> Focused on mission <input checked="" type="checkbox"/> Members know each other  <p>SEM/T 3 / 5</p>	<p>Team</p> <p>Performing</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input type="checkbox"/> Waste continuously eliminated  <p>SEM/T 4 / 5</p>
Trabajo	<p>Work</p> <p>Prepared</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Commitment made <input checked="" type="checkbox"/> Cost and effort estimated <input checked="" type="checkbox"/> Resource availability understood <input checked="" type="checkbox"/> Risk exposure understood <input checked="" type="checkbox"/> Acceptance criteria established <input checked="" type="checkbox"/> Sufficiently broken down to start <input checked="" type="checkbox"/> Tasks identified and prioritized <input checked="" type="checkbox"/> Credible plan in place <input checked="" type="checkbox"/> At least one team member ready <input checked="" type="checkbox"/> Integration points defined  <p>SEM/T 2 / 6</p>	<p>Work</p> <p>Started</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Development started <input checked="" type="checkbox"/> Progress monitored <input checked="" type="checkbox"/> Definition of done in place <input checked="" type="checkbox"/> Tasks being progressed  <p>SEM/T 3 / 6</p>

Way of Working

Way of Working

In Use

- Practices & tools in use
- Regularly inspected
- Adapted to context
- Supported by team
- Feedback mechanisms in place
- Practices & tools support collaboration

SEM/T

3 / 6

Way of Working

In Place

- Used by whole team
- Accessible to whole team
- Inspected and adapted by whole team

SEM/T

4 / 6

Iteración 2

La iteración 2 se llevó adelante entre los días 20 de junio y 04 de julio de 2019.

Objetivos de la iteración

Acorde a lo detallado en la planificación, la segunda iteración del proyecto tuvo los siguientes objetivos:

- Finalizar el desarrollo de la infraestructura necesaria para proveer a la aplicación de un servidor web y una base de datos.
 - Completar el desarrollo de la colección de *endpoints* de las APIs RESTful del servidor web.
- Desarrollar un incremento para el módulo “Asistencia” de la aplicación.
 - Desarrollo de la vista “Toma de Asistencia” dentro del módulo
 - Desarrollo del “generador de códigos QR” como parte del módulo.
 - Desarrollo del “Visualizador de código QR” como parte de la vista de “Perfil” y “Toma de Asistencia”.
 - Desarrollo del servicio “Actividades con Asistencia” como parte del módulo.
 - Desarrollo del servicio “Registro y Lectura QR” como parte del módulo.
- Continuar con las reuniones con el representante de los *stakeholders*.
- Realizar una demostración en vivo al representante de los *stakeholders* del segundo entregable funcional de software.

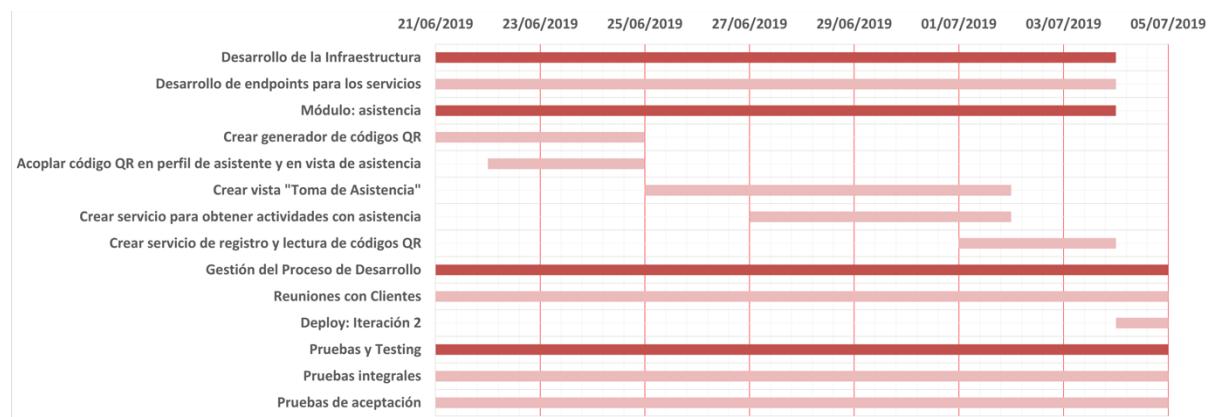


Figura 28 – Recorte de la iteración 2 del diagrama de Gantt del plan de proyecto³².

³²Se puede encontrar en formato digital en

<https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Diagrama%20de%20Gantt/Gantt%20Iteracion%202.png>.

Revisión de los estados alcanzados

<h3>Requirements</h3> <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Acceptable </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	<h3>Software System</h3> <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Demonstrable </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Key architectural characteristics demonstrated <input checked="" type="checkbox"/> System exercised & performance measured <input checked="" type="checkbox"/> Critical HW configurations demonstrated <input checked="" type="checkbox"/> Critical interfaces demonstrated <input checked="" type="checkbox"/> Integration with environment demonstrated <input checked="" type="checkbox"/> Architecture accepted as fit-for-purpose <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 2 / 6  </div>	
<h3>Opportunity</h3> <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Viable </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	<h3>Stakeholders</h3> <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Involved </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Representatives assist the team <input checked="" type="checkbox"/> Timely feedback and decisions provided <input checked="" type="checkbox"/> Changes promptly communicated <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 3 / 6  </div>	
<h3>Team</h3> <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Performing </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input type="checkbox"/> Waste continuously eliminated <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 5  </div>	<h3>Work</h3> <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Started </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Development started <input checked="" type="checkbox"/> Progress monitored <input checked="" type="checkbox"/> Definition of done in place <input checked="" type="checkbox"/> Tasks being progressed <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 3 / 6  </div>	<h3>Way of Working</h3> <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> In Place </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Used by whole team <input checked="" type="checkbox"/> Accessible to whole team <input checked="" type="checkbox"/> Inspected and adapted by whole team <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>

Planificación de la iteración

Realizar la segunda iteración implicó continuar trabajando tanto con el servidor web con el cual se comunica la aplicación móvil y como con la aplicación móvil en sí misma.

Considerando esto, se continuó con la idea de dividir el trabajo relacionado al desarrollo del servidor web y el desarrollo relacionado al incremento de la aplicación web, generando tareas de desarrollo a partir de las historias de usuario concernientes a esta. Luego, las tareas se asignaron a los miembros del equipo.

Se decidió, además, continuar con las pruebas de integración para validar el funcionamiento de las interfaces entre aplicación y servidor.

Alpha “Requerimientos”

The screenshot shows a software interface titled "Requirements". A yellow box labeled "Acceptable" contains the following checklist:

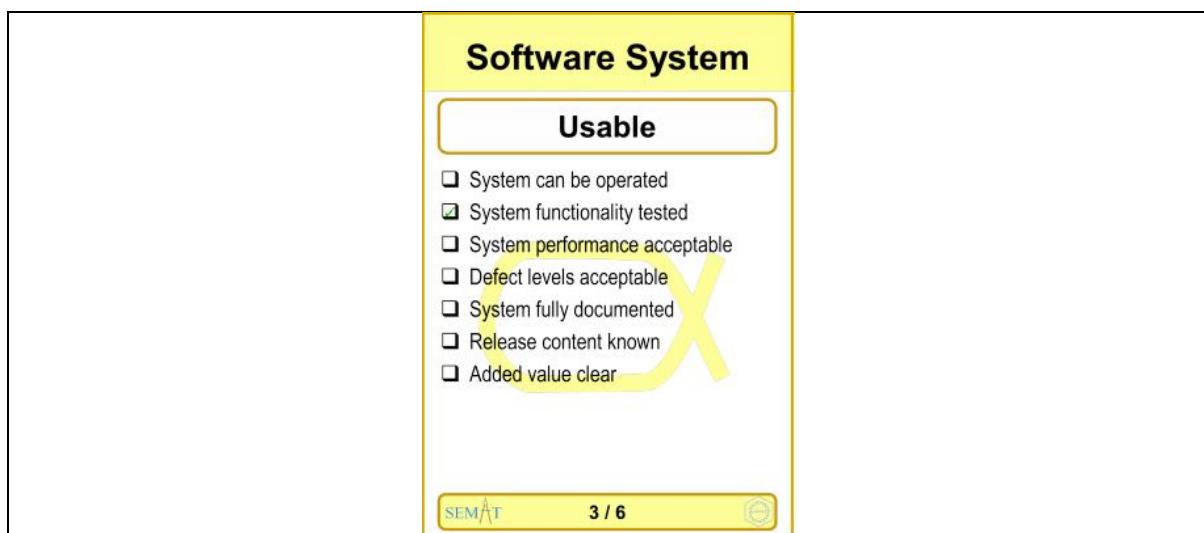
- Acceptable solution described
- Change under control
- Value to be realized clear
- Clear how opportunity is addressed
- Testable

At the bottom of the screen, there is a footer bar with the text "SEM/T" and "4 / 6".

Como resultado de la iteración 2, el *alpha* Requerimiento no reflejó ningún avance en la progresión de estados, quedando en el estado “Aceptable”, cuarto en la escala de seis, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 1.

Alpha “Sistema de Software”



Como resultado de la iteración 2, el *alpha* Sistema de Software reflejó un avance en la progresión de estados, transicionando desde el estado “Demostrable”, segundo en la escala de seis, hacia el estado “Utilizable”, cuyo logro fue abordado parcialmente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

Estado “Utilizable”

El tercer estado de este *alpha* se refiere al Sistema de Software como **Utilizable (Usable)**. Alcanzar este estado implica que el Sistema de Software es utilizable y demuestra todas las características de calidad de un sistema operacional. Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 2*.

Logrados

“System functionality tested (Funcionalidades del sistema probadas)”: Este elemento se considera alcanzado para esta iteración debido a que las funcionalidades del sistema, como parte del desarrollo, son sometidas a pruebas unitarias y de integración previamente a su despliegue en parte de incrementos.

No logrados

“System can be operated (El sistema puede ser operado)”: En esta etapa el sistema todavía no puede ser utilizado por los stakeholders ya que no existe un despliegue de este.

“System performance acceptable (Rendimiento del sistema es aceptable)”: El rendimiento del sistema aún no es aceptable debido a que el desarrollo está en una etapa muy temprana y aún el sistema no está completo, por lo que no se puede considerar a este elemento como cumplido.

“Defect levels acceptable (Niveles de defecto aceptables)”: Debido a que el sistema se encuentra en una etapa temprana de desarrollo y no está completo, no se puede considerar que el nivel de defectos sea el aceptable.

“System fully documented (Sistema documentado completamente)”: En esta etapa del proyecto no existe una documentación del sistema, por lo que no se puede considerar a este elemento como completado.

“Release content known (Contenido del despliegue conocido)”: Si bien para esta etapa el proyecto ya contaba con dos despliegues en su haber, siguiendo lo pautado en la planificación, el equipo decidió marcar este elemento como no alcanzado debido a que los despliegues demostrables aún no revestían un estado enteramente operacional, razón por la cual el contenido de los entregables no estaba disponible para evaluación de posibles usuarios ni del conjunto de los stakeholders por fuera del representante para pruebas en contextos similares a un ambiente de producción.

“Added value clear (Valor agregado claro)”: En esta etapa, al existir solamente dos módulos desarrollados, los stakeholders aún no tienen en claro el valor agregado que produce el sistema, por lo que se considera este requisito como no alcanzado.

Alpha “Oportunidad”



Como resultado de la iteración 2, el *alpha* Oportunidad no reflejó ningún avance en la progresión de estados, quedando en el estado “Viable”, cuarto en la escala de seis, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 0.

Alpha “Stakeholders”



Como resultado de la iteración 2, el *alpha Stakeholders* reflejó un avance en la progresión de estados, transicionando desde el estado “Involucrados”, tercero en la escala de seis, hacia el estado “En acuerdo”, cuyo logro fue abordado parcialmente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

Estado “En acuerdo”

El cuarto estado de este *alpha* se refiere a los *stakeholders* como **En acuerdo (In Agreement)**. Alcanzar este estado implica que el representante de los *stakeholders* está de acuerdo con cómo avanza el progreso de la ejecución del proyecto. Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 2*.

Logrados

“Rep's happy with their involvement (Representante feliz con su involucramiento)”: Se considera cumplido este elemento debido a que, en las reuniones, el cliente se demostró satisfecho y predisposto para colaborar con el equipo de desarrollo.

“Rep's input valued (Aportes del representante valorados)”: Tras las reuniones de feedback y las reuniones periódicas, el equipo logró reaccionar positivamente ante los comentarios del representante, logrando que sienta que sus aportes fueron valorados y respetados, encontrándose sus contribuciones útiles a los fines de aportar a la calidad del producto.

“Team's input valued (Aportes del equipo valorados)”: El representante de los *stakeholders* demostró que los aportes del equipo de desarrollo eran tomados en cuenta y valorados, pudiendo lograr que el resultado de las reuniones sea productivo para ambas partes.

No logrados

“Minimal expectations agreed (Mínimas expectativas acordadas)”: Se considera que la etapa actual es muy temprana como para acordar las mínimas expectativas del representante de los *stakeholders*, por lo cual, se considera que este elemento todavía no puede ser cumplido.

“Priorities clear & perspectives balanced (Prioridades claras y perspectivas balanceadas)”: Como se menciona anteriormente, esta iteración es muy temprana como para cumplir este elemento, por lo que por el momento este elemento queda sin ser alcanzado completamente.

Alpha “Equipo”

The image shows a digital representation of a Scrum board. At the top, a blue bar labeled "Team" spans across the board. Below it, another blue bar labeled "Performing" is highlighted with a thicker border. To the right of the bars, there is a large, stylized blue letter "A". Underneath the bars, a white area contains a bulleted list of five items. Three of the items have a checked box to their left, while the other two have an unchecked box.

Item	Status
Consistently meeting commitments	✓
Continuously adapting to change	✓
Addresses problems	✓
Rework and backtracking minimized	✗
Waste continuously eliminated	✗

SEM/IT 4 / 5

Como resultado de la iteración 2, el *alpha* Equipo no reflejó ningún avance en la progresión de estados, quedando en el estado “Desempeñándose”, cuarto en la escala de cinco, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 1.

Alpha “Trabajo”

Work

Under Control

Tasks being completed
 Unplanned work under control
 Risks under control
 Estimates revised to reflect performance
 Progress measured
 Re-work under control
 Commitments consistently met

SEMÁT4 / 6

Como resultado de la iteración 2, el *alpha* Trabajo reflejó un avance en la progresión de elementos, completando parcialmente el del estado “Bajo control”, cuarto en la escala de seis.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las progresiones en el alcance de los elementos componentes del estado alcanzado.

Estado “Bajo Control”

El cuarto estado de este *alpha*, “**Bajo control**”, refiere a que el trabajo está en buen camino, los riesgos están bajo control y los niveles de productividad son suficientes como para alcanzar un resultado satisfactorio. A continuación, se exponen los elementos alcanzados de y no alcanzados este estado.

Logrados

“Tasks being completed” (Tareas completadas): Debido a que tareas están siendo realizadas en gran medida según lo planificado, se considera a este elemento como alcanzado.

“Estimates revised to reflect performance” (Estimaciones revisadas para reflejar el desempeño): En las reuniones con el representante de los *stakeholders* se hacía una revisión de las estimaciones del trabajo comparando con lo prácticamente realizado, lo que logró reflejar el desempeño del equipo de desarrollo.

“Progress measured” (Progreso medido): Mediante la priorización de los ítems de trabajo y mediante la aplicación del segundo principio de Kanban, el cual refiere a la limitación de trabajo concurrente, el equipo tuvo una manera sencilla de poder verificar el progreso en la ejecución del proyecto, detectando además los posibles *cuellos de botella* que redundan en impedimentos para el desarrollo.

“Commitments consistently met” (Compromisos cumplidos consistentemente): Se puede considerar a este elemento como alcanzado debido a que las tareas planificadas hasta el momento fueron realizadas en tiempo y forma, cumpliendo con los compromisos asumidos con el cliente.

No Logrados

“Unplanned work under control” (Trabajo no planeado bajo control): Debido a que en el momento de esta iteración aún queda mucho trabajo por hacer y el equipo considera que no puede tener completa seguridad acerca de la aparición de trabajo no planificado, se determinó que para esta etapa el trabajo no planeado no puede considerarse como bajo control.

“Risks under control” (Riesgos bajo control): Por el momento, si bien se ha hecho una evaluación de riesgos, al no estar el trabajo no planeado bajo control no se puede considerar a este elemento como cumplido debido a que el nuevo trabajo puede generar nuevos riesgos.

“Re-work under control” (Retrabajo bajo control): Al quedar mucho trabajo por realizar, no se puede afirmar que el retrabajo esté bajo control, por lo que no se considera a este elemento como alcanzado.

Alpha “Way of Working”

The slide has a blue header bar with the text "Way of Working". Below it is a blue box containing the title "Working Well". To the right of the title is a list of four items, each preceded by a checkbox:

- Predictable progresses being made
- Practices naturally applied
- Tools naturally support way-of-working
- Continually tuned

At the bottom left of the slide is the SEMANT logo, and at the bottom center is the text "5 / 6".

Como resultado de la iteración 2, el *alpha* Way of Working reflejó un avance en la progresión de estados, transicionando desde el estado “En Orden”, cuarto en la escala de seis, hacia el estado “Funcionando bien”, cuyo logro fue abordado parcialmente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

Estado “Funcionando Bien”

El quinto estado de este *alpha* se refiere al way of working como **Funcionando Bien (Working Well)**. Alcanzar este estado implica que el way of working adoptado está funcionando bien para el equipo. Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 2*.

Logrados

“Predictable progress being made (Realizando progreso predecible)”: Se puede considerar que este elemento ha sido alcanzado ya que gracias a las prácticas adoptadas en cuanto a la gestión de las tareas se pudo predecir el progreso y así realizar las reuniones con los *stakeholders* con el trabajo planeado en tiempo y forma.

“Practices naturally applied (Prácticas aplicadas naturalmente)”: Mediante la puesta en práctica del primer principio de Kanban, referido a la visibilidad del proceso de desarrollo, tanto el equipo como el representante de los stakeholders podían consultar, de forma transparente, la documentación y el estado actual del proyecto en cualquier momento. Esto permitía, a su vez, la puesta en uso de las prácticas de forma natural y autogestionada.

No logrados

“Tools naturally support way-of-working (Herramientas ayudan al way of working naturalmente)”: En esta etapa del proyecto, considerada por el equipo como temprana para la evaluación de este elemento, se decidió optar por no marcar a este elemento como alcanzado, quedando a la espera de una futura evaluación del mismo luego de lograda una

mayor familiaridad, tanto del equipo como del representante de los stakeholders, del way of working.

“Continually tuned (Calibrado continuamente)”: De forma similar al elemento anterior, el equipo determinó que por encontrarse en una etapa temprana de la ejecución del proyecto, aún no se puede considerar que el way of working haya sido calibrado continuamente, estableciéndose este elemento como no alcanzado.

Retrospectiva

Se describe en esta sección una breve explicación de la transición de estados y de elementos alcanzados como parte de la iteración, producto de la evaluación del equipo del trabajo realizado al final de la iteración 2.

Alpha “Requerimientos”

La iteración comenzó y terminó con un logro parcial del estado de requerimientos **“Aceptables” (Acceptable – 4/6)**, sin modificación alguna en los estados alcanzados. Luego de marcar como logrados varios estados en la evaluación de la *iteración 1*, este *alpha* no marcó progreso alguno durante la segunda iteración.

Alpha “Sistema de Software”

La iteración comenzó con un logro completo del estado de sistema de software **“Demostrable” (Demonstrable – 2/6)** y finalizó con el logro parcial del estado de sistema de software **“Utilizable” (Usable – 3/6)**, marcando progreso hacia el objetivo de permitir que, con parte de su funcionalidad, el sistema de software pueda ser utilizable al final de la iteración por los stakeholders.

Alpha “Oportunidad”

La iteración comenzó y terminó con un logro parcial del estado de oportunidad **“Viable” (Viable – 4/6)**, sin modificación alguna en los estados alcanzados. Luego de marcar como logrados varios estados en la evaluación de la *iteración 0* inicial, este *alpha* no marcó progreso alguno durante la primera y segunda iteración.

Alpha “Stakeholders”

La iteración comenzó con un logro completo del estado de los stakeholders como **“Involucrados” (Involved – 3/6)** dentro de la iniciativa de software y finalizó con un logro parcial del estado **“En acuerdo” (In Agreement – 4/6)**. La progresión de estados se justifica en la notable colaboración desarrollada entre el equipo de desarrollo y el representante de los stakeholders a lo largo de la iteración, demostrándose la colaboración útil para poder llevar adelante el trabajo y dar un feedback de lo que se le ha presentado.

Alpha “Equipo”

La iteración comenzó y finalizó con un logro parcial del estado del equipo como **“Desempeñándose” (Performing – 4/5)**, sin modificación alguna en los estados alcanzados. Luego de marcar como logrados varios estados en la evaluación de la *iteración 1*, este *alpha* no marcó progreso alguno durante la segunda iteración.

Alpha “Trabajo”

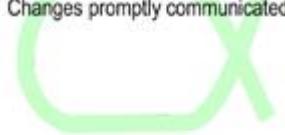
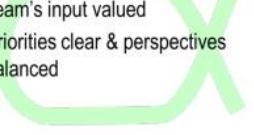
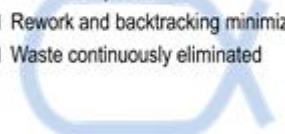
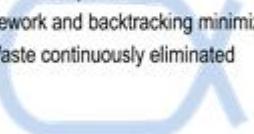
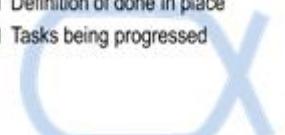
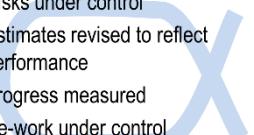
La iteración comenzó con un logro completo del estado del trabajo como **“Comenzado” (Started – 3/6)** y finalizó con un logro parcial del estado del trabajo como **“Bajo control” (Under Control – 4/6)**. La progresión de estados se justifica en la progresión constante del equipo de desarrollo en las tareas establecidas.

Alpha “Way of Working”

La iteración comenzó con un logro completo del estado del way of working como **“En Orden”** (**In Place – 4/6**) y finalizó con un logro parcial del estado del way of working como **“Funcionando bien” (Working Well – 5/6)**. La progresión de estados se justifica en que el trabajo en la iteración 2 permitió continuar la inspección y adaptación del way of working, permitiendo mejorarlo y adaptarlo a las necesidades que fueron surgiendo.

Resumen de cambios de estado durante la iteración

Alpha	Estado Inicial	Estado Final
Requerimientos	<p>Requirements</p> <p>Acceptable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable <p>SEM/T 4 / 6</p>	<p>Requirements</p> <p>Acceptable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable <p>SEM/T 4 / 6</p>
Sistema de Software	<p>Software System</p> <p>Demonstrable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Key architectural characteristics demonstrated <input checked="" type="checkbox"/> System exercised & performance measured <input checked="" type="checkbox"/> Critical HW configurations demonstrated <input checked="" type="checkbox"/> Critical interfaces demonstrated <input checked="" type="checkbox"/> Integration with environment demonstrated <input checked="" type="checkbox"/> Architecture accepted as fit-for-purpose <p>SEM/T 2 / 6</p>	<p>Software System</p> <p>Usable</p> <ul style="list-style-type: none"> <input type="checkbox"/> System can be operated <input checked="" type="checkbox"/> System functionality tested <input type="checkbox"/> System performance acceptable <input type="checkbox"/> Defect levels acceptable <input type="checkbox"/> System fully documented <input type="checkbox"/> Release content known <input type="checkbox"/> Added value clear <p>SEM/T 3 / 6</p>
Oportunidad	<p>Opportunity</p> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <p>SEM/T 4 / 6</p>	<p>Opportunity</p> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <p>SEM/T 4 / 6</p>

<p>Stakeholders</p>	<p>Stakeholders</p> <p>Involved</p> <ul style="list-style-type: none"> <input type="checkbox"/> Representatives assist the team <input checked="" type="checkbox"/> Timely feedback and decisions provided <input checked="" type="checkbox"/> Changes promptly communicated  <p>SEMAT 3 / 6</p>	<p>Stakeholders</p> <p>In Agreement</p> <ul style="list-style-type: none"> <input type="checkbox"/> Minimal expectations agreed <input checked="" type="checkbox"/> Rep's happy with their involvement <input checked="" type="checkbox"/> Rep's input valued <input checked="" type="checkbox"/> Team's input valued <input type="checkbox"/> Priorities clear & perspectives balanced  <p>SEMAT 4 / 6</p>
<p>Equipo</p>	<p>Team</p> <p>Performing</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input type="checkbox"/> Waste continuously eliminated  <p>SEMAT 4 / 5</p>	<p>Team</p> <p>Performing</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input type="checkbox"/> Waste continuously eliminated  <p>SEMAT 4 / 5</p>
<p>Trabajo</p>	<p>Work</p> <p>Started</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Development started <input checked="" type="checkbox"/> Progress monitored <input checked="" type="checkbox"/> Definition of done in place <input checked="" type="checkbox"/> Tasks being progressed  <p>SEMAT 3 / 6</p>	<p>Work</p> <p>Under Control</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Tasks being completed <input type="checkbox"/> Unplanned work under control <input type="checkbox"/> Risks under control <input checked="" type="checkbox"/> Estimates revised to reflect performance <input checked="" type="checkbox"/> Progress measured <input type="checkbox"/> Re-work under control <input checked="" type="checkbox"/> Commitments consistently met  <p>SEMAT 4 / 6</p>

Way of Working

Way of Working

In Place

- Used by whole team
- Accessible to whole team
- Inspected and adapted by whole team



SEM/T

4 / 6

Way of Working

Working Well

- Predictable progresses being made
- Practices naturally applied
- Tools naturally support way-of-working
- Continually tuned

SEM/T

5 / 6

Iteración 3

La iteración 3 se llevó adelante entre los días 5 de julio y 22 de julio de 2019.

Objetivos de la iteración

Acorde a lo detallado en la planificación, la tercera iteración del proyecto tuvo los siguientes objetivos:

- Desarrollar un incremento para el módulo “Notificaciones” de la aplicación.
 - Desarrollo de la vista “Toma de Asistencia” dentro del módulo
 - Desarrollo del “generador de códigos QR” como parte del módulo.
 - Desarrollo del “Visualizador de código QR” como parte de la vista de “Perfil” y “Toma de Asistencia”.
 - Desarrollo del servicio “Actividades con Asistencia” como parte del módulo.
 - Desarrollo del servicio “Registro y Lectura QR” como parte del módulo.
- Desarrollar un incremento para el módulo “Geolocalización” de la aplicación.
 - Configurar la lógica para recibir las notificaciones y mostrarlas en la aplicación.
 - Desarrollo un webapp para el envío de notificaciones personalizadas.
 - Configuración de un proyecto en OneSignal para poder utilizarlo en la aplicación.
- Continuar con las reuniones con el representante de los *stakeholders*.
- Realizar una demostración en vivo al representante de los *stakeholders* del tercer entregable funcional de software.

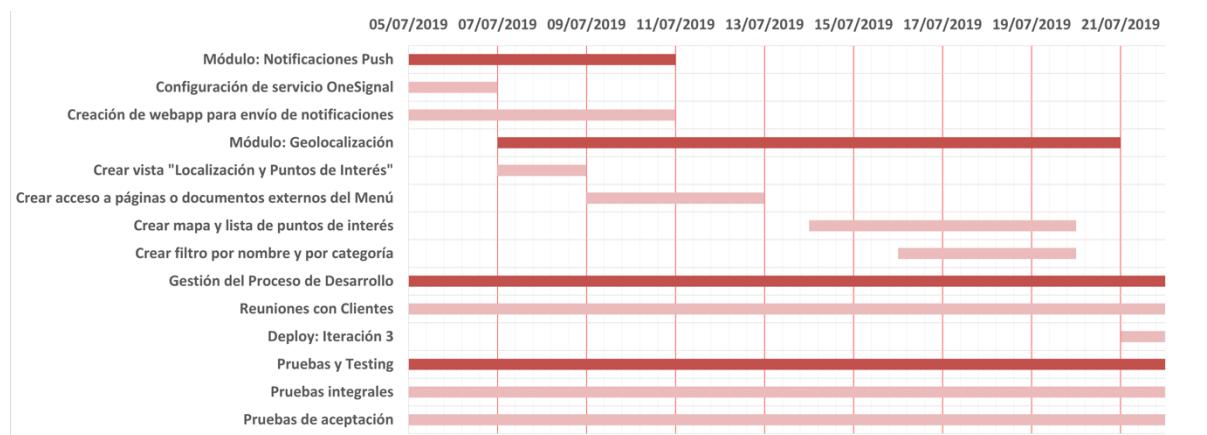


Figura 29 – Recorte de la iteración 3 del diagrama de Gantt del plan de proyecto³³ .

³³Se puede encontrar en formato digital en
<https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Diagrama%20de%20Gantt/Gantt%20Iteracion%203.png>.

Revisión de los estados alcanzados

<h3>Requirements</h3> <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Acceptable </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	<h3>Software System</h3> <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Usable </div> <ul style="list-style-type: none"> <input type="checkbox"/> System can be operated <input checked="" type="checkbox"/> System functionality tested <input type="checkbox"/> System performance acceptable <input type="checkbox"/> Defect levels acceptable <input type="checkbox"/> System fully documented <input type="checkbox"/> Release content known <input type="checkbox"/> Added value clear <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 3 / 6  </div>	
<h3>Opportunity</h3> <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Viable </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	<h3>Stakeholders</h3> <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> In Agreement </div> <ul style="list-style-type: none"> <input type="checkbox"/> Minimal expectations agreed <input checked="" type="checkbox"/> Rep's happy with their involvement <input checked="" type="checkbox"/> Rep's input valued <input checked="" type="checkbox"/> Team's input valued <input type="checkbox"/> Priorities clear & perspectives balanced <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	
<h3>Team</h3> <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Performing </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input type="checkbox"/> Waste continuously eliminated <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 5  </div>	<h3>Work</h3> <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Under Control </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Tasks being completed <input type="checkbox"/> Unplanned work under control <input type="checkbox"/> Risks under control <input checked="" type="checkbox"/> Estimates revised to reflect performance <input checked="" type="checkbox"/> Progress measured <input type="checkbox"/> Re-work under control <input checked="" type="checkbox"/> Commitments consistently met <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	<h3>Way of Working</h3> <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Working Well </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Predictable progresses being made <input checked="" type="checkbox"/> Practices naturally applied <input type="checkbox"/> Tools naturally support way-of-working <input type="checkbox"/> Continually tuned <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 5 / 6  </div>

Planificación de la iteración

Realizar la tercera iteración implicó trabajar con la aplicación móvil y con OneSignal para configurar lo referido al envío de notificaciones, así como el webapp para gestionarlas.

Considerando esto, se continuó generando tareas de desarrollo a partir de las historias de usuario concernientes a esta. Luego, las tareas se asignaron a los miembros del equipo.

También, se continuó con las pruebas de integración para validar el funcionamiento de las interfaces entre aplicación y servidor.

Alpha “Requerimientos”

Requirements

Acceptable

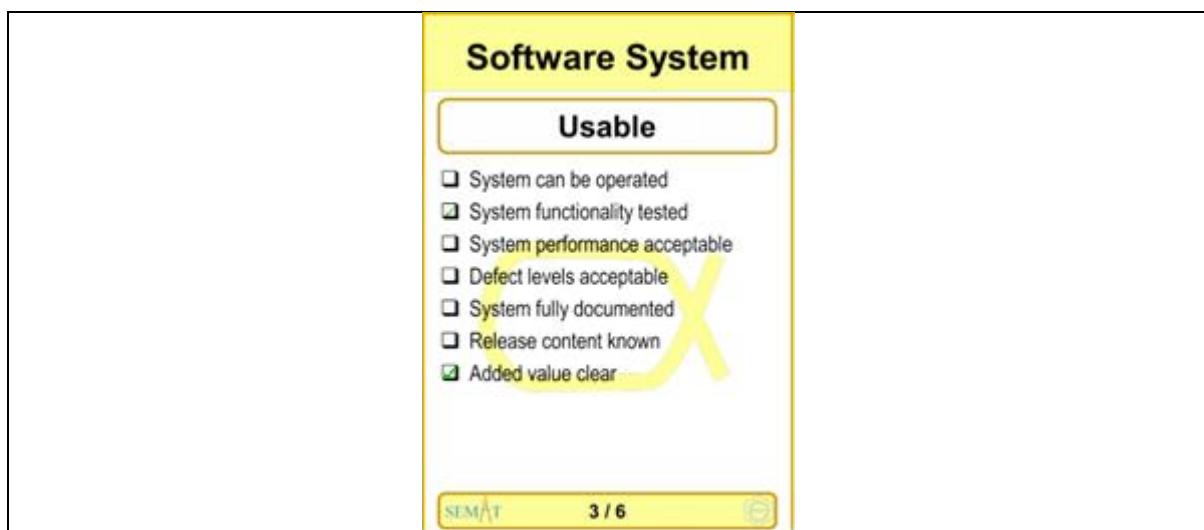
- Acceptable solution described
- Change under control
- Value to be realized clear
- Clear how opportunity is addressed
- Testable

SEM/T **4 / 6** 

Como resultado de la iteración 3, el *alpha* Requerimiento no reflejó ningún avance en la progresión de estados, quedando en el estado “Aceptable”, cuarto en la escala de seis, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 1.

Alpha “Sistema de Software”



Como resultado de la iteración 3, el *alpha* Sistema de Software reflejó un avance en la progresión de un elemento en el estado “Utilizable”, tercero en la escala de seis.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar el alcance del elemento modificado. Los elementos no alcanzados continúan fundamentándose en las razones detalladas en la iteración 2.

El tercer estado de este *alpha* se refiere al Sistema de Software como **Utilizable (Usable)**. Alcanzar este estado implica que el Sistema de Software es utilizable y demuestra todas las características de calidad de un sistema operacional. Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 2*.

Logrados

“Added value clear” (Valor agregado claro): A través de las reuniones y el feedback de los stakeholders, se ha concluido que los mismos entienden el valor agregado que aporta el sistema, por lo que se puede considerar que este elemento ha sido alcanzado.

Alpha “Oportunidad”



Como resultado de la iteración 3, el *alpha* Oportunidad no reflejó ningún avance en la progresión de estados, quedando en el estado “Viable”, cuarto en la escala de seis, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado ya no se fundamentan en las razones de la iteración 0, sino que ahora se fundamentan en que el financiamiento ya no va a ser posible debido a que los *stakeholders* decidieron utilizar una aplicación diferente para el evento.

Alpha “Stakeholders”

The screenshot shows a digital interface for managing stakeholder interactions. The title "Stakeholders" is at the top, followed by a section titled "In Agreement". Below this are five items, each with a checkbox:

- Minimal expectations agreed
- Rep's happy with their involvement
- Rep's input valued
- Team's input valued
- Priorities clear & perspectives balanced

A large green checkmark is drawn over the entire list. At the bottom right, it says "4 / 6".

Como resultado de la iteración 3, el *alpha Stakeholders* no reflejó ningún avance en la progresión de estados, quedando en el estado “En acuerdo”, cuarto en la escala de seis, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 2.

Alpha “Equipo”

The image shows a digital scrum board for the 'Alpha "Equipo"' team. At the top, a blue header bar says 'Team'. Below it, a white box labeled 'Performing' is highlighted with a blue border. To the right of this box is a large, stylized blue letter 'A' representing the team name. Under the 'Performing' box is a list of five items, each preceded by a checkbox:

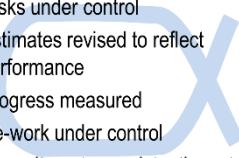
- Consistently meeting commitments
- Continuously adapting to change
- Addresses problems
- Rework and backtracking minimized
- Waste continuously eliminated

At the bottom left of the board, there is a small blue button labeled 'SEM/IT'. In the bottom right corner of the board area, there is a progress bar with the number '4 / 5' in the center.

Como resultado de la iteración 3, el *alpha* Equipo no reflejó ningún avance en la progresión de estados, quedando en el estado “Desempeñándose”, cuarto en la escala de cinco, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 1.

Alpha “Trabajo”

Work
Under Control
<input checked="" type="checkbox"/> Tasks being completed <input type="checkbox"/> Unplanned work under control <input type="checkbox"/> Risks under control <input checked="" type="checkbox"/> Estimates revised to reflect performance <input checked="" type="checkbox"/> Progress measured <input type="checkbox"/> Re-work under control <input checked="" type="checkbox"/> Commitments consistently met

SEMANTIC
4 / 6


Como resultado de la iteración 3, el *alpha* Trabajo no reflejó ningún avance en la progresión de estados, quedando en el estado “Bajo control”, cuarto en la escala de seis, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 2.

Alpha “Way of Working”

Way of Working

Working Well

- Predictable progresses being made
- Practices naturally applied
- Tools naturally support way-of-working
- Continually tuned



SEM | T5 / 6...

Como resultado de la iteración 3, el *alpha* Way of Working reflejó un avance en la progresión de estados, completando el estado “Funcionando bien” (Working Well).

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

Estado “Funcionando Bien”

El quinto estado de este *alpha* se refiere al *way of working* como **Funcionando bien (Working Well)**. Alcanzar este estado implica que el *way of working* adoptado está funcionando bien para el equipo. Este estado fue completamente alcanzado por el equipo en la revisión realizada como parte de la *iteración 3*.

“Tools naturally support way-of-working (Herramientas ayudan al way of working naturalmente)”: Luego de la ejecución de esta iteración, el equipo determinó que las herramientas hasta aquí utilizadas soportaron su *way of working*, sin requerir el uso de herramientas extra para la gestión del proceso, de la metodología, de las prácticas o del proyecto. Las herramientas que fueron utilizadas, sin cambios, son las que fueron detalladas tanto en la planificación del presente proyecto como así también en el presente informe.

“Continually tuned (Calibrado continuamente)”: Se puede considerar que a partir el uso del *way of working* hasta el momento, el mismo se ha comenzado a calibrar continuamente según las necesidades del equipo, por lo que en esta etapa se consideró a este elemento como alcanzado.

Retrospectiva

Se describe en esta sección una breve explicación de la transición de estados y de elementos alcanzados como parte de la iteración, producto de la evaluación del equipo del trabajo realizado al final de la iteración 3.

Alpha “Requerimientos”

La iteración comenzó y terminó con un logro parcial del estado de requerimientos **“Aceptables” (Acceptable – 4/6)**, sin modificación alguna en los estados alcanzados. Luego de marcar como logrados varios estados en la evaluación de la *iteración 1*, este *alpha* no marcó progreso alguno durante la tercera iteración.

Alpha “Sistema de Software”

La iteración comenzó con un logro parcial del estado de sistema de software **“Utilizable” (Usable – 3/6)**, y finalizó con el logro parcial del estado de sistema de software **“Utilizable” (Usable – 3/6)** agregando solo un elemento como cumplido, marcando progreso hacia el objetivo de permitir que, con parte de su funcionalidad, el sistema de software pueda ser utilizable al final de la iteración por los stakeholders.

Alpha “Oportunidad”

La iteración comenzó y terminó con un logro parcial del estado de oportunidad **“Viable” (Viable – 4/6)**, sin modificación alguna en los estados alcanzados. En esta iteración, la razón de su estancamiento pasa a ser la decisión por parte de los *stakeholders* de usar otra aplicación en su evento.

Alpha “Stakeholders”

La iteración comenzó y terminó con un logro parcial del estado de los stakeholders como **“En acuerdo” (In Agreement – 4/6)**, sin modificación alguna en los estados alcanzados. Luego de marcar como logrados varios estados en la evaluación de la *iteración 2*, este *alpha* no marcó progreso alguno durante la tercera iteración.

Alpha “Equipo”

La iteración comenzó y finalizó con un logro parcial del estado del equipo como **“Desempeñándose” (Performing – 4/5)**, sin modificación alguna en los estados alcanzados. Luego de marcar como logrados varios estados en la evaluación de la *iteración 1*, este *alpha* no marcó progreso alguno durante la tercera iteración.

Alpha “Trabajo”

La iteración comenzó y terminó con un logro parcial del estado del trabajo como **“Bajo control” (Under Control – 4/6)**, sin modificación alguna en los estados alcanzados. Luego de marcar como logrados varios estados en la evaluación de la *iteración 2*, este *alpha* no marcó progreso alguno durante la tercera iteración.

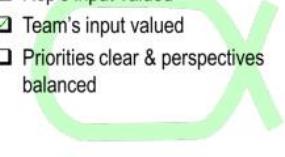
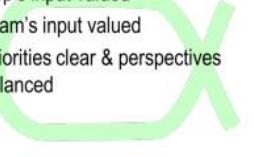
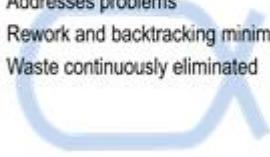
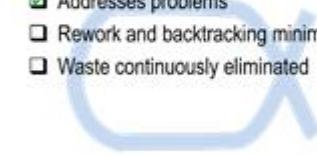
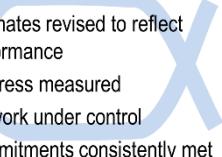
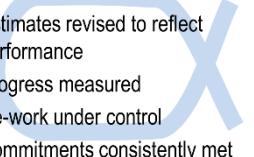
Alpha “Way of Working”

La iteración comenzó con un logro parcial del estado del way of working como **“Funcionando bien” (Working Well – 5/6)** y finalizó con un logro completo del estado del way of working como **“Funcionando bien” (Working Well – 5/6)**. La progresión de estados se justifica en

que el trabajo en la iteración 3 permitió continuar la inspección y adaptación del way of working, permitiendo mejorarlo y adaptarlo a las necesidades que fueron surgiendo.

Cambios en los estados durante la iteración

Alpha	Estado Inicial	Estado Final
Requerimientos	<p>Requirements</p> <p>Acceptable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable <p>SEM/T 4 / 6</p>	<p>Requirements</p> <p>Acceptable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable <p>SEM/T 4 / 6</p>
Sistema de Software	<p>Software System</p> <p>Usable</p> <ul style="list-style-type: none"> <input type="checkbox"/> System can be operated <input checked="" type="checkbox"/> System functionality tested <input type="checkbox"/> System performance acceptable <input type="checkbox"/> Defect levels acceptable <input type="checkbox"/> System fully documented <input type="checkbox"/> Release content known <input checked="" type="checkbox"/> Added value clear <p>SEM/T 3 / 6</p>	<p>Software System</p> <p>Usable</p> <ul style="list-style-type: none"> <input type="checkbox"/> System can be operated <input type="checkbox"/> System functionality tested <input type="checkbox"/> System performance acceptable <input type="checkbox"/> Defect levels acceptable <input type="checkbox"/> System fully documented <input type="checkbox"/> Release content known <input checked="" type="checkbox"/> Added value clear <p>SEM/T 3 / 6</p>
Oportunidad	<p>Opportunity</p> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <p>SEM/T 4 / 6</p>	<p>Opportunity</p> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <p>SEM/T 4 / 6</p>

<p>Stakeholders</p>	<p>Stakeholders</p> <p>In Agreement</p> <ul style="list-style-type: none"> <input type="checkbox"/> Minimal expectations agreed <input checked="" type="checkbox"/> Rep's happy with their involvement <input checked="" type="checkbox"/> Rep's input valued <input checked="" type="checkbox"/> Team's input valued <input type="checkbox"/> Priorities clear & perspectives balanced  <p>SEMAT 4 / 6</p>	<p>Stakeholders</p> <p>In Agreement</p> <ul style="list-style-type: none"> <input type="checkbox"/> Minimal expectations agreed <input checked="" type="checkbox"/> Rep's happy with their involvement <input checked="" type="checkbox"/> Rep's input valued <input checked="" type="checkbox"/> Team's input valued <input type="checkbox"/> Priorities clear & perspectives balanced  <p>SEMAT 4 / 6</p>
<p>Equipo</p>	<p>Team</p> <p>Performing</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input type="checkbox"/> Waste continuously eliminated  <p>SEMAT 4 / 5</p>	<p>Team</p> <p>Performing</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input type="checkbox"/> Waste continuously eliminated  <p>SEMAT 4 / 5</p>
<p>Trabajo</p>	<p>Work</p> <p>Under Control</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Tasks being completed <input type="checkbox"/> Unplanned work under control <input type="checkbox"/> Risks under control <input checked="" type="checkbox"/> Estimates revised to reflect performance <input checked="" type="checkbox"/> Progress measured <input type="checkbox"/> Re-work under control <input checked="" type="checkbox"/> Commitments consistently met  <p>SEMAT 4 / 6</p>	<p>Work</p> <p>Under Control</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Tasks being completed <input type="checkbox"/> Unplanned work under control <input type="checkbox"/> Risks under control <input checked="" type="checkbox"/> Estimates revised to reflect performance <input checked="" type="checkbox"/> Progress measured <input type="checkbox"/> Re-work under control <input checked="" type="checkbox"/> Commitments consistently met  <p>SEMAT 4 / 6</p>

Way of Working

Way of Working

Working Well

- Predictable progresses being made
- Practices naturally applied
- Tools naturally support way-of-working
- Continually tuned

Way of Working

Working Well

- Predictable progresses being made
- Practices naturally applied
- Tools naturally support way-of-working
- Continually tuned

SEM/T

5 / 6

SEM/T

5 / 6

Iteración 4

La iteración 4 se llevó adelante entre los días 22 de julio y 30 de julio de 2019.

Objetivos de la iteración

Acorde a lo detallado en la planificación, la cuarta iteración del proyecto tuvo los siguientes objetivos:

- Desarrollar un incremento para el módulo “Moderadores, Chairs y Organizadores” de la aplicación.
 - Desarrollo de la vista “Chairs” dentro del módulo
 - Desarrollo de la vista “Comité organizador” dentro del módulo
 - Desarrollo de la vista “Detalle de moderador, chair u organizador” dentro del módulo
 - Desarrollo del servicio “Obtener Moderadores, Chairs y Organizadores” como parte del módulo.
- Continuar con las reuniones con el representante de los *stakeholders*.
- Realizar una demostración en vivo al representante de los *stakeholders* del cuarto entregable funcional de software.
- Desplegar el sistema en un ambiente de pruebas para que los *stakeholders* puedan verlo y probarlo.



Figura 30 – Recorte de la iteración 4 del diagrama de Gantt del plan de proyecto³⁴ .

³⁴Se puede encontrar en formato digital en

<https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Diagrama%20de%20Gantt/Gantt%20Iteracion%204.png>.

Revisión de los estados alcanzados

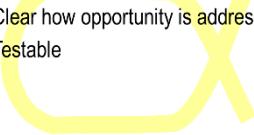
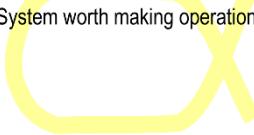
<h3>Requirements</h3> <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Acceptable </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	<h3>Software System</h3> <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Usable </div> <ul style="list-style-type: none"> <input type="checkbox"/> System can be operated <input type="checkbox"/> System functionality tested <input type="checkbox"/> System performance acceptable <input type="checkbox"/> Defect levels acceptable <input type="checkbox"/> System fully documented <input type="checkbox"/> Release content known <input checked="" type="checkbox"/> Added value clear <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 3 / 6  </div>	
<h3>Opportunity</h3> <div style="background-color: #e0ffe0; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Viable </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <div style="background-color: #e0ffe0; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	<h3>Stakeholders</h3> <div style="background-color: #e0ffe0; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> In Agreement </div> <ul style="list-style-type: none"> <input type="checkbox"/> Minimal expectations agreed <input checked="" type="checkbox"/> Rep's happy with their involvement <input checked="" type="checkbox"/> Rep's input valued <input checked="" type="checkbox"/> Team's input valued <input type="checkbox"/> Priorities clear & perspectives balanced <div style="background-color: #e0ffe0; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	
<h3>Team</h3> <div style="background-color: #e0e0ff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Performing </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input type="checkbox"/> Waste continuously eliminated <div style="background-color: #e0e0ff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 5  </div>	<h3>Work</h3> <div style="background-color: #e0e0ff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Under Control </div> <ul style="list-style-type: none"> <input type="checkbox"/> Tasks being completed <input type="checkbox"/> Unplanned work under control <input type="checkbox"/> Risks under control <input type="checkbox"/> Estimates revised to reflect performance <input type="checkbox"/> Progress measured <input type="checkbox"/> Re-work under control <input checked="" type="checkbox"/> Commitments consistently met <div style="background-color: #e0e0ff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6  </div>	<h3>Way of Working</h3> <div style="background-color: #e0e0ff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Working Well </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Predictable progresses being made <input checked="" type="checkbox"/> Practices naturally applied <input checked="" type="checkbox"/> Tools naturally support way-of-working <input checked="" type="checkbox"/> Continually tuned <div style="background-color: #e0e0ff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 5 / 6  </div>

Planificación de la iteración

Realizar la cuarta iteración implicó ejecutar trabajo correspondiente a funcionalidades de la aplicación móvil. Tomando esto en consideración, se continuó generando tareas de desarrollo a partir de las historias de usuario correspondientes a la aplicación móvil. Luego, las tareas se asignaron a los miembros del equipo.

Como en las iteraciones previas, se continuó con las pruebas de integración para validar el funcionamiento de las interfaces entre aplicación y servidor.

Alpha “Requerimientos”

Requirements	Requirements	Requirements
Acceptable	Addressed	Fulfilled
<p><input checked="" type="checkbox"/> Acceptable solution described <input checked="" type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable</p> 	<p><input checked="" type="checkbox"/> Enough addressed to be acceptable <input checked="" type="checkbox"/> Requirements and system match <input checked="" type="checkbox"/> Value realized clear <input checked="" type="checkbox"/> System worth making operational</p> 	<p><input checked="" type="checkbox"/> Stakeholders accept requirements <input checked="" type="checkbox"/> No hindering requirements <input checked="" type="checkbox"/> Requirements fully satisfied</p> 
SEM <small> </small> T 4 / 6	SEM <small> </small> T 5 / 6	SEM <small> </small> T 6 / 6

Como resultado de la iteración 4, el *alpha* Requerimientos reflejó un avance en la progresión de estados, transicionando desde el estado **Aceptable (Acceptable)**, cuarto en la escala de seis, hacia el estado **Completados (Fulfilled)**, cuyo logro abordado completamente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

Estado “Aceptados”

El cuarto estado de este *alpha*, Requerimientos “**Aceptados**”, refiere a que los requerimientos describen un sistema que es aceptable para los *stakeholders*.

“Change under control” (Cambios bajo control): Considerando que en esta iteración se finalizaron las tareas vinculadas a programación, se puede considerar que los requerimientos del sistema no presentarán cambios mayores y los mismos se encuentran bajo control, por lo que se puede afirmar que este elemento ha sido alcanzado.

Estado “Abordados”

El quinto estado de este *alpha*, Requerimientos “**Abordados**”, refiere a que una cantidad aceptable de requerimientos ha sido abordada para satisfacer las necesidades del nuevo sistema en una forma que es aceptable por los *stakeholders*.

“Enough addressed to be acceptable” (Suficientes abordados para ser aceptable): A través de las reuniones realizadas con el representante de los *stakeholders* y el posterior refinamiento de los requerimientos en tareas reflejadas en el tablero de Trello, se consideró que los requerimientos estaban suficientemente abordados como para que el sistema sea aceptable.

“Requirements and system match” (Requerimientos y sistema coinciden): Gracias a las reuniones de demostración con el representante de los stakeholders y su feedback se determinó que los requerimientos y el sistema coinciden.

“Value realized clear” (Valor de realizado claro): Al tener el sistema desarrollado enteramente, al representante de los stakeholders le queda claro el valor de realizar el sistema.

“System worth making operational” (Sistema rentable de hacerse operacional): Se ha determinado que la implementación de los requerimientos del sistema es rentable y aceptable para hacerlo operacional.

Estado “Cumplidos”

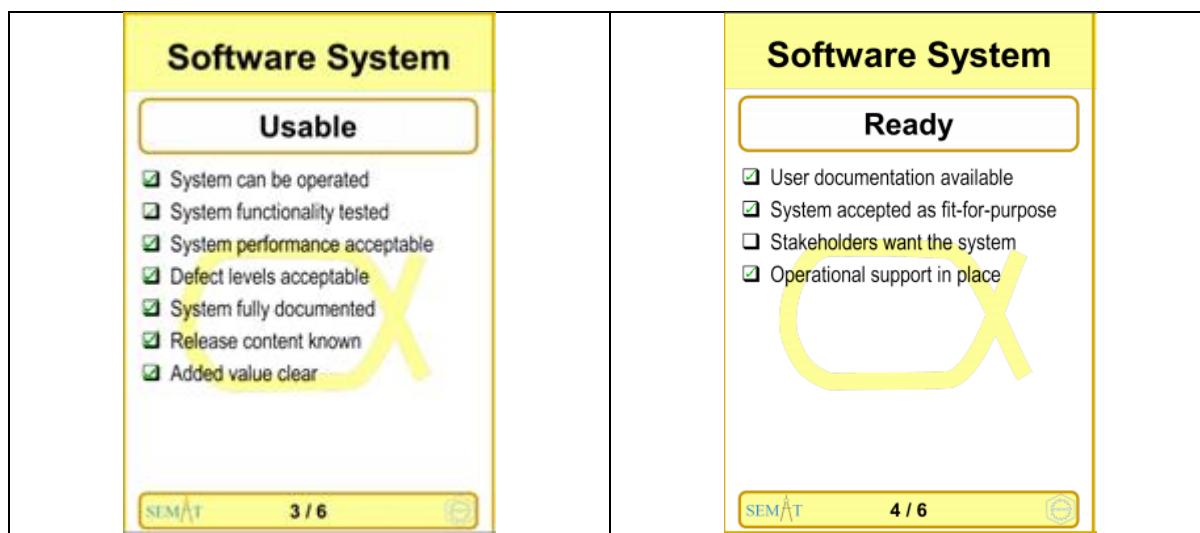
El sexto estado de este *alpha*, Requerimientos “**Cumplidos**”, refiere a que los requerimientos que han sido abordados satisfacen completamente la necesidad del sistema.

“Stakeholders accept requirements” (Stakeholders aceptan los requerimientos): El representante de los stakeholders ha demostrado que los requerimientos lograron capturar lo que requerían para satisfacer la necesidad que cubre el sistema. Por este motivo, se puede considerar que este elemento ha sido alcanzado.

“No hindering requirements” (Sin requerimientos que obstaculicen): En esta etapa todos los requerimientos han sido abordados y cumplidos, por lo que se puede considerar que no quedaron requerimientos pendientes que impidan que el representante de los stakeholders acepte el sistema.

“Requirements fully satisfied” (Requerimientos completamente satisfechos): Todos los requerimientos fueron abordados y satisfechos en tiempo y forma, dejando la columna de Backlog y Pendientes de Programar del tablero de Trello vacías, por lo que se puede considerar que este elemento ha sido cumplido.

Alpha “Sistema de Software”



Como resultado de la iteración 4, el *alpha* Sistema de Software reflejó un avance en la progresión de estados, transicionando desde el estado “Utilizable”, tercero en la escala de seis, hacia el estado “Listo”, cuyo logro fue abordado completamente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar el alcance del elemento modificado.

Estado “Utilizable”

El tercer estado de este *alpha* se refiere al Sistema de Software como **Utilizable (Usable)**. Alcanzar este estado implica que el Sistema de Software es utilizable y demuestra todas las características de calidad de un sistema operacional. Este estado fue completamente alcanzado por el equipo en la revisión realizada como parte de la *iteración 4*.

“System can be operated” (El sistema puede ser operado): En esta etapa se provee a los stakeholders de un ambiente en el que pueden realizar pruebas sobre el sistema, por lo que se puede considerar a este elemento como alcanzado.

“System functionality tested” (Funcionalidad del sistema probada): A través de las reuniones y pruebas de integración, unitarias y de aceptación, se ha concluido que las funcionalidades del sistema están probadas, por lo que se puede considerar que este elemento ha sido alcanzado.

“System performance acceptable” (Rendimiento del sistema aceptable): Se puede demostrar a través del ambiente de pruebas que el rendimiento del sistema es aceptable para los stakeholders, considerando este elemento como cumplido.

“Defect levels acceptable” (Niveles de defecto aceptables): Las pruebas unitarias, de aceptación e integrales arrojaron un nivel de defectos aceptable por los stakeholders, marcando este elemento como alcanzado.

“Release content known” (Contenido del despliegue conocido): A través de las reuniones, las pruebas de aceptación y el ambiente de pruebas que se le brindó a los stakeholders, se ha concluido que los mismos son conscientes del contenido que se va a publicar, por lo que se puede considerar que este elemento ha sido alcanzado.

“System fully documented” (Sistema completamente documentado): En cuanto a la documentación sobre la usabilidad de aplicación, se podría considerar que esta autodocumentada debido al diseño intuitivo de sus interfaces que sigue las líneas de diseño correspondiente a las plataformas donde se utilizaría (Android e iOS). Además, en cuanto al diseño interno de la aplicación, se han presentado diagramas de distinto tipo que sirven como documentación. Considerando esto, se puede considerar al elemento como alcanzado.

Estado “Listo”

El cuarto estado de este *alpha* se refiere al Sistema de Software como **Listo (Ready)**. Alcanzar este estado implica que el Sistema de Software ha sido aceptado para un despliegue en un ambiente productivo. Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 4*.

Logrados

“User documentation available” (Documentación para el usuario disponible): Se ha proporcionado documentación a los stakeholders sobre la instalación y uso del sistema, por lo que se puede considerar a este elemento como completado.

“System accepted as fit-for-purpose” (Sistema aceptado como apto para el propósito): A través de las reuniones y el feedback de los stakeholders, se ha concluido que el sistema es apto para el propósito que tiene, satisfaciendo las necesidades que debería cumplir. Por este motivo, se puede considerar que este elemento ha sido alcanzado.

“Operational support in place” (Soporte operativo establecido): El equipo de desarrollo actuó como soporte operativo para los stakeholders en las pruebas realizadas tanto en conjunto como por separado en el ambiente de pruebas. Por esto mismo se puede considerar a este elemento como cumplido.

No Logrados

“Stakeholders want the system” (Los stakeholders quieren el sistema): Los stakeholders decidieron optar por otra aplicación para su evento, por lo que en esta etapa el equipo se encontró con la situación de perder la fuente de financiamiento para el desarrollo. Como consecuencia, además, los stakeholders decidieron que el sistema no esté operacional, favoreciendo una solución de software diferente a la desarrollada en el presente proyecto. Por este motivo se puede considerar que este elemento no fue alcanzado; permaneciendo así para todo el tiempo restante del desarrollo del proyecto y bloqueando el progreso de los estados del alpha Sistema de Software.

Alpha “Oportunidad”

The screenshot shows a digital board titled "Opportunity". A green box labeled "Viable" contains a checklist with six items, all of which have been checked off with a green checkmark. The items are:

- Solution outlined
- Solution possible within constraints
- Risks acceptable & manageable
- Solution profitable
- Reasons to develop solution understood
- Pursuit viable

At the bottom left of the board, it says "SEM/T". At the bottom center, it says "4 / 6".

Como resultado de la iteración 4, el *alpha* Oportunidad reflejó un avance en la progresión de un elemento en el “Viable”, cuarto en la escala de seis.

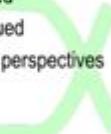
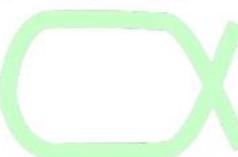
Se detallan a continuación las evaluaciones realizadas por el equipo para considerar el alcance del elemento modificado. Los elementos no alcanzados continúan fundamentándose en las razones detalladas en la iteración 3.

Estado “Viable”

El cuarto estado de este *alpha* se refiere a la oportunidad como **Viable**. Esto significa que además de estar identificado el valor, se suman a ellos una aproximación de cómo se abordará la solución de software, juntamente con la identificación de cómo se gestionarán los riesgos que implica el llevar adelante la iniciativa. Este estado fue parcialmente alcanzado por el equipo en la revisión realizada como parte de la *iteración 4*.

“Solution possible within constraints (Solución posible dentro de las restricciones identificadas)”: En esta iteración, se ha logrado controlar el trabajo no planeado, por lo partiendo del trabajo aplicado a la gestión de riesgos del trabajo planificado, y considerando los nuevos riesgos presentados por el trabajo no planeado, se puede considerar que la solución es posible dentro de las restricciones identificadas.

Alpha "Stakeholders"

	<p>Stakeholders</p> <p>In Agreement</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Minimal expectations agreed <input type="checkbox"/> Rep's happy with their involvement <input type="checkbox"/> Rep's input valued <input type="checkbox"/> Team's input valued <input checked="" type="checkbox"/> Priorities clear & perspectives balanced  <p>SEMAT 4 / 6</p>		<p>Stakeholders</p> <p>Satisfied for Deployment</p> <ul style="list-style-type: none"> <input type="checkbox"/> Stakeholder feedback provided <input checked="" type="checkbox"/> System ready for deployment  <p>SEMAT 5 / 6</p>
--	--	--	---

Como resultado de la iteración 4, el *alpha Stakeholders* reflejó un avance en la progresión de estados, transicionando desde el estado “En acuerdo”, cuarto en la escala de seis, hacia el estado “Satisfechos para despliegue del software”, cuyo logro fue abordado completamente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las progresiones en el alcance de los elementos componentes del estado alcanzado.

Estado “En acuerdo”

El cuarto estado de este *alpha* se refiere a los *stakeholders* como **En acuerdo (In Agreement)**. Alcanzar este estado implica que el representante de los *stakeholders* está de acuerdo con cómo va el proyecto. Este estado fue completamente alcanzado por el equipo en la revisión realizada como parte de la *iteración 4*.

“Minimal expectations agreed (Mínimas expectativas acordadas)”: Se considera que, en esta iteración, en la que el desarrollo ha sido finalizado, el representante de los *stakeholders* pudo hacer un análisis del sistema y se acordaron las mínimas expectativas para el despliegue del sistema.

“Priorities clear & perspectives balanced (Prioridades claras y perspectivas balanceadas)”: En este momento, con el sistema desarrollado, el representante de los *stakeholders* junto con el equipo de desarrollo, determinaron cuáles serían las prioridades y perspectivas para proveer una dirección clara para el equipo.

El cuarto estado de este *alpha* se refiere a los *stakeholders* como **Satisfechos para despliegue del software (Satisfied for Deployment)**. Alcanzar este estado implica que se han cumplido las mínimas expectativas del representante de los *stakeholders*. Este estado fue completamente alcanzado por el equipo en la revisión realizada como parte de la *iteración 4*.

“Stakeholder feedback provided (Stakeholder proveyó feedback)”: Luego de las reuniones y pruebas de aceptación, se realizó una reunión en la que el representante de los stakeholders proveyó un feedback al equipo de desarrollo desde el punto de vista de un stakeholders. Por este motivo se puede considerar a este elemento como cumplido.

“System ready for deployment (Sistema listo para el despliegue)”: Luego de las reuniones y pruebas de aceptación, el representante de los stakeholders ha confirmado que el sistema está listo para su despliegue. Se considera a este elemento como alcanzado.

Alpha “Equipo”



Como resultado de la iteración 4, el *alpha* Equipo reflejó un avance en la progresión de estados, completando el estado “Desempeñándose”.

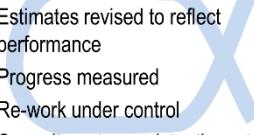
Se detallan a continuación las evaluaciones realizadas por el equipo para considerar el alcance de los elementos componentes del estado.

El cuarto estado de este *alpha* refiere a la capacidad del equipo de encontrarse **Desempeñándose**. Alcanzar este estado implica que quienes integran el equipo ya están trabajando de forma efectiva y eficiente. A partir de la iteración anterior, se mencionan los nuevos requisitos logrados.

“Rework and backtracking minimized” (Retrabajo y retroceso minimizado): En la actual iteración, se considera a este elemento como cumplido debido a que tanto el equipo de desarrollo como el representante de los stakeholders se han logrado adaptar al proyecto y consiguieron mejorar este aspecto, por lo que se puede considerar a este elemento como alcanzado.

“Waste continuously eliminated” (Desperdicio continuo eliminado): En esta iteración, al haber recibido feedback constante del representante de los stakeholders y al haber desarrollado la mayor parte del sistema, se considera que el desperdicio de trabajo es casi nulo. Por esta razón, se considera a este elemento como cumplido.

Alpha “Trabajo”

Work	Work
Under Control  <ul style="list-style-type: none"><input checked="" type="checkbox"/> Tasks being completed<input checked="" type="checkbox"/> Unplanned work under control<input checked="" type="checkbox"/> Risks under control<input checked="" type="checkbox"/> Estimates revised to reflect performance<input checked="" type="checkbox"/> Progress measured<input checked="" type="checkbox"/> Re-work under control<input checked="" type="checkbox"/> Commitments consistently met	Concluded  <ul style="list-style-type: none"><input checked="" type="checkbox"/> Only admin tasks left<input checked="" type="checkbox"/> Results achieved<input checked="" type="checkbox"/> Resulting system accepted
 4 / 6 	 5 / 6 

Como resultado de la iteración 4, el *alpha* reflejó un avance en la progresión de estados, transicionando desde el estado “Bajo Control”, cuarto en la escala de seis, hacia el estado “Concluido”, cuyo logro fue abordado completamente

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las progresiones en el alcance de los elementos componentes del estado alcanzado.

Estado “Bajo control”

El cuarto estado de este *alpha*, “**Bajo control**”, refiere a que el trabajo está en buen camino, los riesgos están bajo control y los niveles de productividad son suficientes como para alcanzar un resultado satisfactorio. A continuación, se exponen los elementos alcanzados de este estado.

“Unplanned work under control” (Trabajo no planeado bajo control): Debido a que las tareas de desarrollo han sido finalizadas en esta iteración, el trabajo no planeado es mínimo o inexistente, por este motivo se puede considerar a este elemento como alcanzado.

“Risks under control” (Riesgos bajo control): Al haber logrado controlar el trabajo no planeado, los riesgos también pudieron ser controlados debido a que estos están relacionados. Por ello en esta etapa el equipo consideró al elemento como alcanzado.

“Re-work under control” (Retrabajo bajo control): El retrabajo ha sido controlado gracias a las pruebas de aceptación con el representante de los stakeholders, por lo que, al validar el trabajo realizado, el retrabajo fue minimizado. Dadas estas condiciones, el equipo consideró para esta etapa a este elemento como cumplido.

Estado “Concluido”

El quinto estado de este *alpha*, “**Concluido**”, refiere a que el trabajo referido a la generación de valor en el producto desarrollado en el proyecto ha sido concluido. A continuación, se detallan los elementos alcanzados de este estado.

“Only admin tasks left” (Solo restan tareas de administrador): Debido a que el trabajo referido a programación e implementación ha sido completado en esta iteración, sólo restan tareas que son principalmente tareas de administración, como ser el data entry o la configuración de los ambientes de despliegue. Por este motivo se considera al elemento como alcanzado.

“Results achieved” (Resultados alcanzados): Se han logrado alcanzar los resultados esperados, habiendo sido validados por el representante de los stakeholders a través de pruebas de aceptación y el despliegue de un ambiente de pruebas.

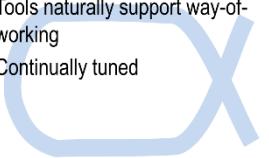
“Resulting system accepted” (Sistema resultante aceptado): Al haber terminado el desarrollo del sistema en esta etapa, el proyecto ya no demanda del esfuerzo del equipo de desarrollo.

Alpha “Way of Working”

Way of Working

Working Well

- Predictable progresses being made
- Practices naturally applied
- Tools naturally support way-of-working
- Continually tuned



SEM | T5 / 6

Como resultado de la iteración 4, el *alpha* Way of Working no reflejó ningún avance en la progresión de estados, quedando en el estado “Funcionando bien”, quinto en la escala de seis, cuyo logro fue abordado completamente.

Retrospectiva

Se describe en esta sección una breve explicación de la transición de estados y de elementos alcanzados como parte de la iteración, producto de la evaluación del equipo del trabajo realizado al final de la iteración 4.

Alpha “Requerimientos”

La iteración comenzó con un logro parcial del estado de requerimientos **“Aceptables” (Acceptable – 4/6)** y finalizó con un logro completo del estado de requerimientos **“Cumplidos” (Fulfilled – 6/6)**, visualizándose un marcado progreso en los estados de este *Alpha*. Para lograr esta marcada progresión se destaca el alto grado de colaboración entre el equipo de desarrollo y el representante de los stakeholders.

Alpha “Sistema de Software”

La iteración comenzó con un logro parcial del estado de sistema de software **“Utilizable” (Usable – 3/6)**, y finalizó con el logro parcial del estado de sistema de software **“Listo” (Ready – 4/6)**, marcando progreso hacia el objetivo de permitir que el sistema de software pueda ser estar listo para el despliegue.

Alpha “Oportunidad”

La iteración comenzó y terminó con un logro parcial del estado de oportunidad **“Viable” (Viable – 4/6)**, con la modificación de un elemento del estado. En esta iteración, la razón de su estancamiento continúa siendo la decisión por parte de los *stakeholders* de usar otra aplicación en su evento.

Alpha “Stakeholders”

La iteración comenzó con un logro parcial del estado de los stakeholders **“En acuerdo” (In Agreement – 4/6)**, y finalizó con el logro completo del estado de los stakeholders **“Satisfechos para despliegue del software” (Satisfied for Deployment – 5/6)**. La progresión de estados se justifica en la notable colaboración desarrollada entre el equipo de desarrollo y el representante de los stakeholders a lo largo de la iteración, demostrándose la colaboración útil para poder llevar adelante el trabajo y dar un feedback de lo que se le ha presentado.

Alpha “Equipo”

La iteración comenzó con un logro parcial del estado del equipo como **“Desempeñándose” (Performing – 4/5)**, y finalizó con el logro completo del mismo. La progresión obedeció a la adaptación del equipo a trabajar con el objetivo en mente de trabajar como una única unidad, mantener el cumplimiento de los compromisos detallados en la planificación y continuar con la generación de entregables de valor producto del trabajo.

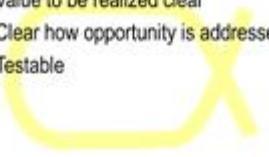
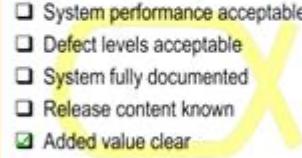
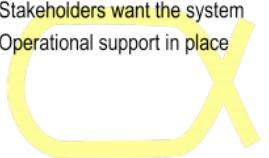
Alpha “Trabajo”

La iteración comenzó con un logro parcial del estado del trabajo como **“Bajo control” (Under Control – 4/6)**, y finalizó con el logro completo del estado del trabajo como **“Concluido” (Concluded – 5/6)**. La progresión de estados se justifica en la progresión constante del equipo de desarrollo en las tareas establecidas.

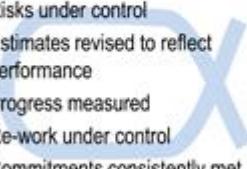
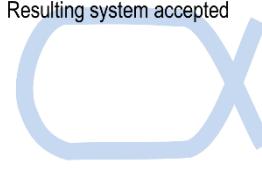
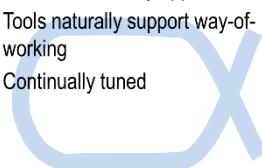
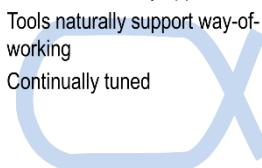
Alpha "Way of Working"

La iteración comenzó y terminó con un logro completo del estado del way of working como **"Funcionando bien"** (**Working Well – 5/6**), sin modificación alguna en los estados alcanzados. Luego de marcar como logrados varios estados en la evaluación de la *iteración 3*, este *alpha* no marcó progreso alguno durante la cuarta iteración.

Resumen de cambios de estado durante la iteración

Alpha	Estado Inicial	Estado Final
Requerimientos	<p>Requirements</p> <p>Acceptable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Acceptable solution described <input type="checkbox"/> Change under control <input checked="" type="checkbox"/> Value to be realized clear <input checked="" type="checkbox"/> Clear how opportunity is addressed <input checked="" type="checkbox"/> Testable  <div style="background-color: #f0f0f0; padding: 5px; display: flex; justify-content: space-between;"> SEM/IT 4 / 6 </div>	<p>Requirements</p> <p>Fulfilled</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Stakeholders accept requirements <input checked="" type="checkbox"/> No hindering requirements <input checked="" type="checkbox"/> Requirements fully satisfied  <div style="background-color: #f0f0f0; padding: 5px; display: flex; justify-content: space-between;"> SEM/IT 6 / 6 </div>
Sistema de Software	<p>Software System</p> <p>Usable</p> <ul style="list-style-type: none"> <input type="checkbox"/> System can be operated <input checked="" type="checkbox"/> System functionality tested <input type="checkbox"/> System performance acceptable <input type="checkbox"/> Defect levels acceptable <input type="checkbox"/> System fully documented <input type="checkbox"/> Release content known <input checked="" type="checkbox"/> Added value clear  <div style="background-color: #f0f0f0; padding: 5px; display: flex; justify-content: space-between;"> SEM/IT 3 / 6 </div>	<p>Software System</p> <p>Ready</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> User documentation available <input checked="" type="checkbox"/> System accepted as fit-for-purpose <input type="checkbox"/> Stakeholders want the system <input checked="" type="checkbox"/> Operational support in place  <div style="background-color: #f0f0f0; padding: 5px; display: flex; justify-content: space-between;"> SEM/IT 4 / 6 </div>

Oportunidad	<h3>Opportunity</h3> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <p>SEM/T 4 / 6</p>	<h3>Opportunity</h3> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable <p>SEM/T 4 / 6</p>
Stakeholders	<h3>Stakeholders</h3> <p>In Agreement</p> <ul style="list-style-type: none"> <input type="checkbox"/> Minimal expectations agreed <input checked="" type="checkbox"/> Rep's happy with their involvement <input checked="" type="checkbox"/> Rep's input valued <input checked="" type="checkbox"/> Team's input valued <input type="checkbox"/> Priorities clear & perspectives balanced <p>SEM/T 4 / 6</p>	<h3>Stakeholders</h3> <p>Satisfied for Deployment</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Stakeholder feedback provided <input checked="" type="checkbox"/> System ready for deployment <p>SEM/T 5 / 6</p>
Equipo	<h3>Team</h3> <p>Performing</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input type="checkbox"/> Rework and backtracking minimized <input checked="" type="checkbox"/> Waste continuously eliminated <p>SEM/T 4 / 5</p>	<h3>Team</h3> <p>Performing</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input checked="" type="checkbox"/> Rework and backtracking minimized <input checked="" type="checkbox"/> Waste continuously eliminated <p>SEM/T 4 / 5</p>

<p>Trabajo</p>	<p>Work</p> <p>Under Control</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Tasks being completed <input type="checkbox"/> Unplanned work under control <input type="checkbox"/> Risks under control <input checked="" type="checkbox"/> Estimates revised to reflect performance <input checked="" type="checkbox"/> Progress measured <input type="checkbox"/> Re-work under control <input checked="" type="checkbox"/> Commitments consistently met  <p>SEMAT 4 / 6</p>	<p>Work</p> <p>Concluded</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Only admin tasks left <input checked="" type="checkbox"/> Results achieved <input checked="" type="checkbox"/> Resulting system accepted  <p>SEMAT 5 / 6</p>
<p>Way of Working</p>	<p>Way of Working</p> <p>Working Well</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Predictable progresses being made <input checked="" type="checkbox"/> Practices naturally applied <input checked="" type="checkbox"/> Tools naturally support way-of-working <input checked="" type="checkbox"/> Continually tuned  <p>SEMAT 5 / 6</p>	<p>Way of Working</p> <p>Working Well</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Predictable progresses being made <input checked="" type="checkbox"/> Practices naturally applied <input checked="" type="checkbox"/> Tools naturally support way-of-working <input checked="" type="checkbox"/> Continually tuned  <p>SEMAT 5 / 6</p>

Iteración 5

La iteración 5 se llevó adelante entre los días 30 de julio y 07 de agosto de 2019.

Objetivos de la iteración

Acorde a lo detallado en la planificación, la quinta iteración del proyecto tuvo los siguientes objetivos:

- Cargar la información relacionada con el evento en la base de datos
- Configurar la aplicación con los estilos específicos para el evento
- Continuar con las reuniones con el representante de los *stakeholders*.
- Hacer una revisión final con los stakeholders

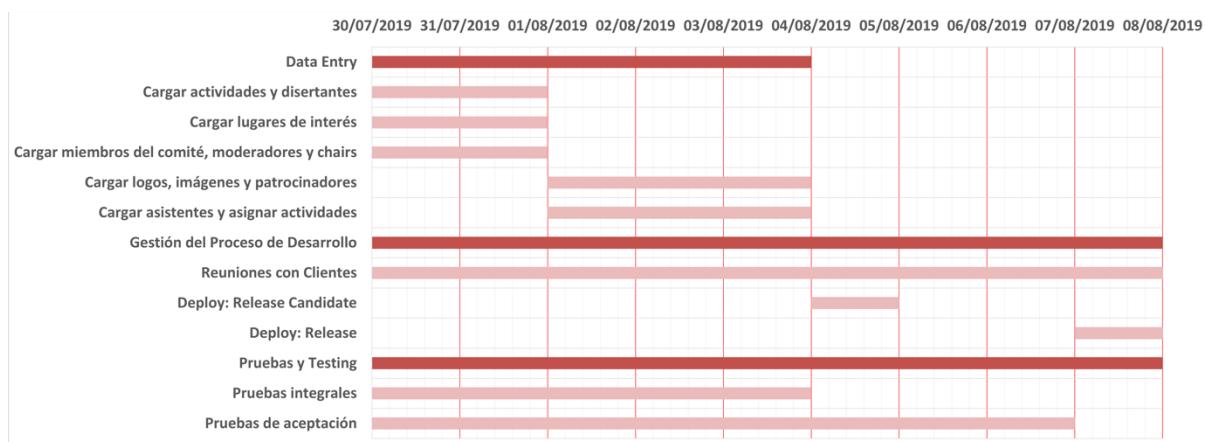
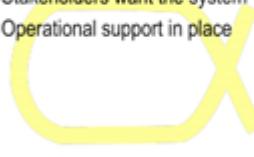
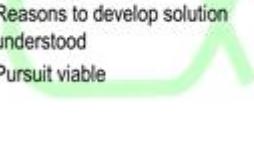
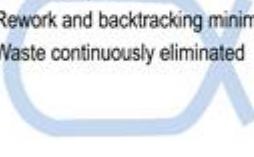
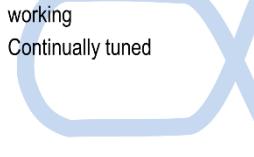


Figura 31 – Recorte de la iteración 5 del diagrama de Gantt del plan de proyecto³⁵.

³⁵Se puede encontrar en formato digital en <https://github.com/fedealbertengo/Conferentia/blob/master/Informe/Diagrama%20de%20Gantt/Gantt%20Iteracion%205.png>.

Revisión de los estados alcanzados

<h3>Requirements</h3> <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Fulfilled </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Stakeholders accept requirements <input checked="" type="checkbox"/> No hindering requirements <input checked="" type="checkbox"/> Requirements fully satisfied  <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 6 / 6 ↻ </div>	<h3>Software System</h3> <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Ready </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> User documentation available <input checked="" type="checkbox"/> System accepted as fit-for-purpose <input checked="" type="checkbox"/> Stakeholders want the system <input checked="" type="checkbox"/> Operational support in place  <div style="background-color: #ffffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6 ↻ </div>	
<h3>Opportunity</h3> <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Viable </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input checked="" type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input checked="" type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable  <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 6 ↻ </div>	<h3>Stakeholders</h3> <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Satisfied for Deployment </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Stakeholder feedback provided <input checked="" type="checkbox"/> System ready for deployment  <div style="background-color: #ccffcc; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 5 / 6 ↻ </div>	
<h3>Team</h3> <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Performing </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input checked="" type="checkbox"/> Rework and backtracking minimized <input checked="" type="checkbox"/> Waste continuously eliminated  <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 4 / 5 ↻ </div>	<h3>Work</h3> <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Concluded </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Only admin tasks left <input checked="" type="checkbox"/> Results achieved <input checked="" type="checkbox"/> Resulting system accepted  <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 5 / 6 ↻ </div>	<h3>Way of Working</h3> <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-bottom: 10px;"> Working Well </div> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Predictable progresses being made <input checked="" type="checkbox"/> Practices naturally applied <input checked="" type="checkbox"/> Tools naturally support way-of-working <input checked="" type="checkbox"/> Continually tuned  <div style="background-color: #ccccff; border: 1px solid black; padding: 5px; margin-top: 10px;"> SEMAT 5 / 6 ↻ </div>

Planificación de la iteración

Realizar la quinta iteración implicó cargar los datos relacionados con el evento y configurar los estilos de la aplicación.

Considerando esto se dividieron las tareas entre los miembros del equipo y el representante de los stakeholders.

También, se continuó con las pruebas de aceptación con los stakeholders.

Alpha “Requerimientos”

Requirements

Fulfilled

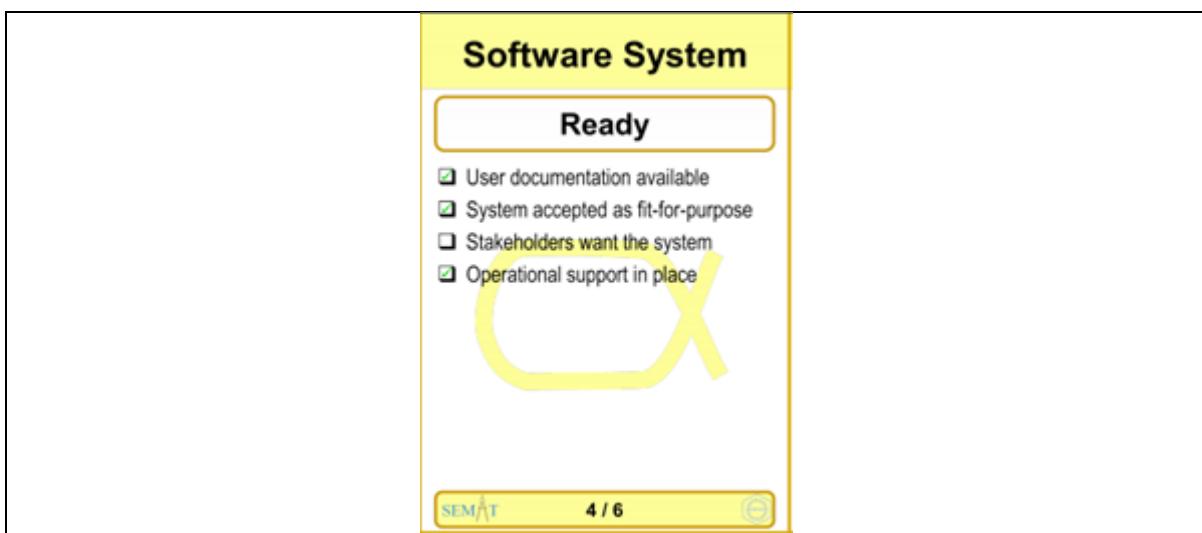
- Stakeholders accept requirements
- No hindering requirements
- Requirements fully satisfied

OK

SEM/T 6 / 6

Como resultado de la iteración 5, el *alpha* Requerimiento no reflejó ningún avance en la progresión de estados, quedando en el estado “Cumplidos”, sexto en la escala de seis, cuyo logro fue abordado completamente.

Alpha “Sistema de Software”



Como resultado de la iteración 5, el *alpha* Sistema de Software no reflejó ningún avance en la progresión de estados, quedando en el estado “Listo”, cuarto en la escala de seis, cuyo logro fue abordado completamente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 4.

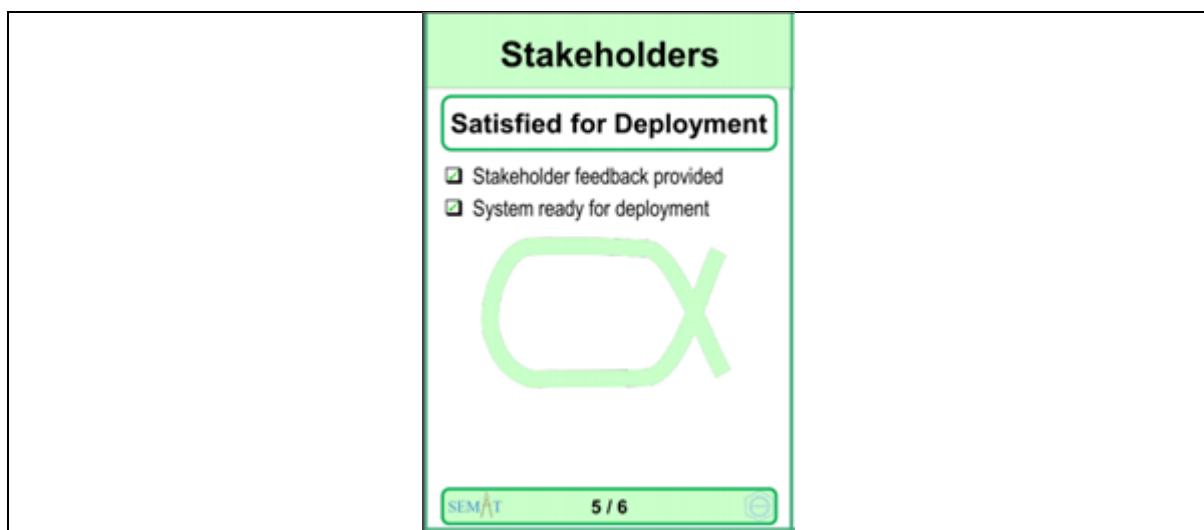
Alpha “Oportunidad”



Como resultado de la iteración 5, el *alpha* Oportunidad no reflejó ningún avance en la progresión de estados, quedando en el estado “Viable”, cuarto en la escala de seis, cuyo logro fue abordado parcialmente.

Las razones del logro parcial de este estado continúan fundamentándose en las razones detalladas en la iteración 2.

Alpha “Stakeholders”



Como resultado de la iteración 5, el *alpha* Stakeholders no reflejó ningún avance en la progresión de estados, quedando en el estado “Satisfechos para despliegue del software”, cuarto en la escala de seis, cuyo logro fue abordado parcialmente.

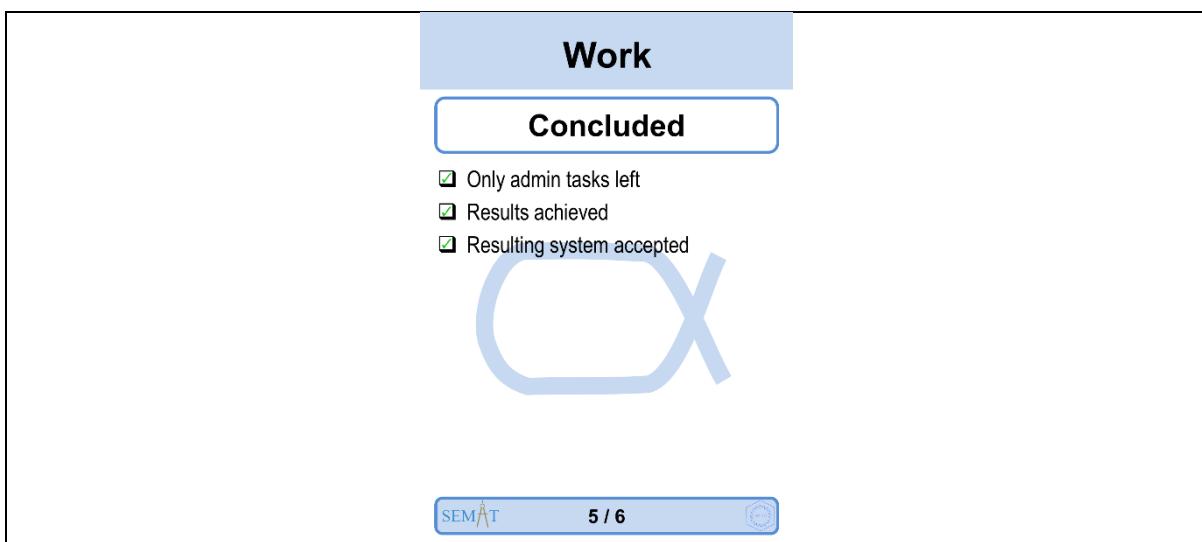
Las razones del logro parcial de este estado continúan fundamentándose en que el próximo estado implicaría que el sistema esté en uso.

Alpha “Equipo”



Como resultado de la iteración 5, el *alpha* Equipo no reflejó ningún avance en la progresión de estados, quedando en el estado “Desempeñándose”, cuarto en la escala de cinco, cuyo logro fue abordado completamente.

Alpha “Trabajo”



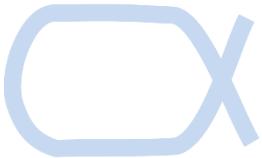
Como resultado de la iteración 5, el *alpha* Trabajo no reflejó ningún avance en la progresión de estados, quedando en el estado “Concluido”, quinto en la escala de seis, cuyo logro fue abordado completamente.

Alpha “Way of Working”

Way of Working

Retired

No longer in use
 Lessons learned shared



SEM | T 6 / 6 

Como resultado de la iteración 5, el *alpha Way of Working* reflejó un avance en la progresión de estados, transicionando desde el estado “Funcionando bien”, quinto en la escala de seis, hacia el estado “Retirado”, cuyo logro fue abordado completamente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

El sexto estado de este *alpha* se refiere al *way of working* como **Retirado (Retired)**. Alcanzar este estado implica que el *way of working* adoptado ya no es utilizado por el equipo. Este estado fue completamente alcanzado por el equipo en la revisión realizada como parte de la *iteración 5*.

“No longer in use (Fuera de uso)”: Al haber finalizado el trabajo y no tener en vista un proyecto adicional al cual adaptar el *way of working* generado, el equipo ya no necesita del *way of working*, por lo que el mismo queda fuera de uso.

“Lessons learned shared (Lecciones aprendidas compartidas)”: Al final de la iteración se realizó una reunión en la que tanto el equipo de desarrollo como el representante de los stakeholders compartieron un feedback sobre el *way of working* utilizado.

Retrospectiva

Se describe en esta sección una breve explicación de la transición de estados y de elementos alcanzados como parte de la iteración, producto de la evaluación del equipo del trabajo realizado al final de la iteración 4.

Alpha “Requerimientos”

La iteración comenzó y terminó con un logro completo del estado de oportunidad **“Cumplidos” (Fulfilled – 6/6)**, sin modificación alguna en los estados alcanzados. Esto es debido a que el Alpha ya ha sido completado.

Alpha “Sistema de Software”

La iteración comenzó y terminó con un logro parcial del estado de sistema de software **“Listo” (Ready – 4/6)**, sin modificación alguna en los estados alcanzados. Esto es debido a que el elemento “Stakeholders want the system” no puede avanzar debido a que los stakeholders han optado por otra aplicación para su evento.

Alpha “Oportunidad”

La iteración comenzó y terminó con un logro parcial del estado de oportunidad **“Viable” (Viable – 4/6)**, sin modificación alguna en los estados alcanzados. En esta iteración, la razón de su estancamiento pasa a ser la decisión por parte de los *stakeholders* de usar otra aplicación en su evento.

Alpha “Stakeholders”

La iteración comenzó y con el logro completo del estado de los stakeholders **“Satisfechos para despliegue del software” (Satisfied for Deployment – 5/6)**, sin modificación alguna en los estados alcanzados. Esto es debido a que el próximo estado implicaría que el sistema esté en uso.

Alpha “Equipo”

La iteración comenzó y finalizó con el logro completo del estado del equipo como **“Desempeñándose” (Performing – 4/5)**, sin modificación alguna en los estados alcanzados. Es así porque aún falta que el equipo realice una retrospectiva final antes de disolverse.

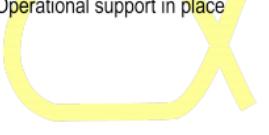
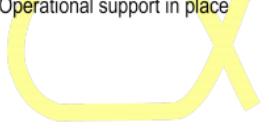
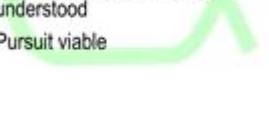
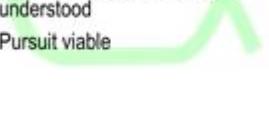
Alpha “Trabajo”

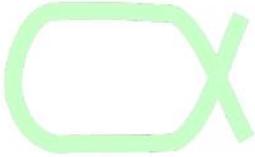
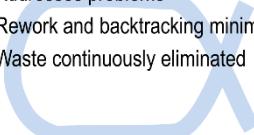
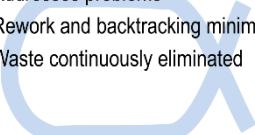
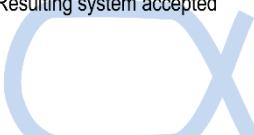
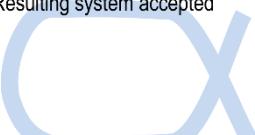
La iteración comenzó y finalizó con el logro completo del estado del trabajo como **“Concluido” (Concluded – 5/6)**, sin modificación alguna en los estados alcanzados.

Alpha “Way of Working”

La iteración comenzó completo del estado del way of working como **“Funcionando bien” (Working Well – 5/6)**, y finalizó con el logro parcial del estado de way of working como **“Retirado” (Retired – 6/6)**, debido a que el equipo ya ha concluido con su aplicación.

Resumen de cambios de estado durante la iteración

Alpha	Estado Inicial	Estado Final
Requerimientos	<p>Requirements</p> <p>Fulfilled</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Stakeholders accept requirements <input checked="" type="checkbox"/> No hindering requirements <input checked="" type="checkbox"/> Requirements fully satisfied  <p>SEMANT 6 / 6</p>	<p>Requirements</p> <p>Fulfilled</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Stakeholders accept requirements <input checked="" type="checkbox"/> No hindering requirements <input checked="" type="checkbox"/> Requirements fully satisfied  <p>SEMANT 6 / 6</p>
Sistema de Software	<p>Software System</p> <p>Ready</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> User documentation available <input checked="" type="checkbox"/> System accepted as fit-for-purpose <input type="checkbox"/> Stakeholders want the system <input checked="" type="checkbox"/> Operational support in place  <p>SEMANT 4 / 6</p>	<p>Software System</p> <p>Ready</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> User documentation available <input checked="" type="checkbox"/> System accepted as fit-for-purpose <input type="checkbox"/> Stakeholders want the system <input checked="" type="checkbox"/> Operational support in place  <p>SEMANT 4 / 6</p>
Oportunidad	<p>Opportunity</p> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable  <p>SEMANT 4 / 6</p>	<p>Opportunity</p> <p>Viable</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Solution outlined <input type="checkbox"/> Solution possible within constraints <input checked="" type="checkbox"/> Risks acceptable & manageable <input type="checkbox"/> Solution profitable <input checked="" type="checkbox"/> Reasons to develop solution understood <input checked="" type="checkbox"/> Pursuit viable  <p>SEMANT 4 / 6</p>

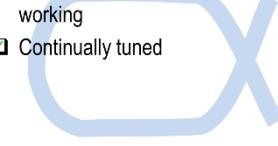
Stakeholders	<p>Stakeholders</p> <p>Satisfied for Deployment</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Stakeholder feedback provided <input checked="" type="checkbox"/> System ready for deployment  <p>SEMAT 5 / 6</p>	<p>Stakeholders</p> <p>Satisfied for Deployment</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Stakeholder feedback provided <input checked="" type="checkbox"/> System ready for deployment  <p>SEMAT 5 / 6</p>
Equipo	<p>Team</p> <p>Performing</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input checked="" type="checkbox"/> Rework and backtracking minimized <input checked="" type="checkbox"/> Waste continuously eliminated  <p>SEMAT 4 / 5</p>	<p>Team</p> <p>Performing</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Consistently meeting commitments <input checked="" type="checkbox"/> Continuously adapting to change <input checked="" type="checkbox"/> Addresses problems <input checked="" type="checkbox"/> Rework and backtracking minimized <input checked="" type="checkbox"/> Waste continuously eliminated  <p>SEMAT 4 / 5</p>
Trabajo	<p>Work</p> <p>Concluded</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Only admin tasks left <input checked="" type="checkbox"/> Results achieved <input checked="" type="checkbox"/> Resulting system accepted  <p>SEMAT 5 / 6</p>	<p>Work</p> <p>Concluded</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Only admin tasks left <input checked="" type="checkbox"/> Results achieved <input checked="" type="checkbox"/> Resulting system accepted  <p>SEMAT 5 / 6</p>

Way of Working

Way of Working

Working Well

- Predictable progresses being made
- Practices naturally applied
- Tools naturally support way-of-working
- Continually tuned



SEMATM

5 / 6



Way of Working

Retired

- No longer in use
- Lessons learned shared



SEMATM

6 / 6



Retrospectiva final

Al final de todas las iteraciones, el equipo decidió hacer una retrospectiva final sobre los resultados de la gestión del proyecto, analizando los distintos estados alcanzados de los alphas y en caso de bloqueo de alguno, su razón.

Sin embargo, no todos los alphas quedaron estáticos luego de la iteración 5, aunque se ha finalizado el desarrollo y configuración del producto, algunos alphas hicieron un último avance en sus estados, los cuales se detallan a continuación.

Alpha “Equipo”

Team

Adjourned

Responsibilities fulfilled
 Members available to other teams
 Mission concluded

OK

SEMÁT 5 / 5

Como resultado de la retrospectiva final, el *alpha* Equipo reflejó un avance en la progresión de estados, transicionando desde el estado “Performando”, cuarto en la escala de cinco, hacia el estado “Disuelto”, cuyo logro fue abordado completamente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las transiciones de estado y el alcance de los elementos componentes de cada estado.

Estado “Disuelto”

El quinto estado de este *alpha* refiere al equipo como **Disuelto**. Alcanzar este estado implica que quienes integran el equipo ya no están dedicados al proyecto y el equipo ha sido disuelto.

“Responsibilities fulfilled (Responsibilidades cumplidas)”: Cada miembro del equipo ha cumplido las responsabilidades que habían asumido.

“Members available to other teams (Miembros disponibles para otros equipos)”: Los miembros del equipo quedan liberados de las responsabilidades del equipo actual y están disponibles para ser asignados a otros equipos de trabajo.

“Mission concluded (Misión concluida)”: Al haber terminado el desarrollo y configuración del sistema, la misión se considera concluida y ya no requiere que el equipo ponga esfuerzo para completarla.

Alpha “Trabajo”

Work

Closed

- Lessons learned
- Metrics available
- Everything archived
- Budget reconciled & closed
- Team released
- No outstanding, uncompleted tasks

SEMANTIC

6 / 6

Como resultado de la retrospectiva final, el *alpha* reflejó un avance en la progresión de estados, transicionando desde el estado “Completado”, quinto en la escala de seis, hacia el estado “Cerrado”, cuyo logro fue abordado parcialmente.

Se detallan a continuación las evaluaciones realizadas por el equipo para considerar las progresiones en el alcance de los elementos componentes del estado alcanzado.

Estado “Cerrado”

El sexto estado de este *alpha*, “**Cerrado**”, refiere a que todas las tareas restantes fueron completadas y el trabajo ha sido oficialmente cerrado. A continuación, se exponen los elementos alcanzados y no alcanzados de este estado.

Logrados

“Lessons learned” (Lecciones aprendidas): En la reunión de retrospectiva final, se han analizado las lecciones aprendidas tanto por el equipo de desarrollo como por el representante de los stakeholders.

“Metrics available” (Métricas disponibles): Las métricas resultantes del sistema fueron expuestas al representante de los stakeholders demostrando el trabajo logrado.

“Everything archived” (Todo logrado): Todas las tareas que estaban en las columnas “Backlog” y “Pendientes de Programar” del tablero de Trello han sido finalizadas con éxito, por lo que se considera a este elemento como cumplido.

“Team released” (Equipo liberado): Al haber terminado todo el trabajo, el equipo de desarrollo y los stakeholders se consideraron liberados del presente proyecto.

“No outstanding, uncompleted tasks” (Sin tareas pendientes restantes ni tareas incompletas): Ninguna tarea, tanto las planificadas como las no planificadas, quedó pendiente ni incompleta.

No Logrados

“Budget reconciled & closed” (Presupuesto conciliado y cerrado): Al no haber pagado por parte de los stakeholders, ya que decidieron no adquirir el sistema de software, no se puede considerar que este elemento haya sido alcanzado.

Revisión y análisis de los estados alcanzados

A continuación, como parte de la retrospectiva final del proyecto, se incluye un análisis de los estados finales alcanzados para cada alpha al momento de concluir la ejecución del proyecto. Se explica, en cada caso, por qué se ha llegado a determinado nivel de alcance de cada estado y se explica, de ser el caso, por qué algunos alphas han quedado bloqueados en un estado determinado sin posibilidad de progresar hasta el estado final.

Alpha “Requerimientos”

El estado final alcanzado para el alpha Requerimientos es “**Cumplidos**” (**Fulfilled – 6/6**), con todos sus elementos alcanzados.

Haber logrado abordar todos los estados de este alpha obedece tanto a la pericia del equipo de desarrollo para implementar los requerimientos definidos para el proyecto como también a la predisposición del equipo y del representante de los stakeholders para realizar reuniones periódicas y de aceptación, brindando feedback constantemente sobre el sistema de software desarrollado.

Alpha “Sistema de Software”

El estado final alcanzado para el alpha Sistema de Software fue “**Listo**” (**Ready – 4/6**), alcanzado parcialmente sin estar cumplido el elemento “*Los stakeholders quieren el sistema*”.

El motivo de que no haya sido posible alcanzar todos los estados de este alpha obedece a que los stakeholders decidieron optar por otro sistema diferente para la gestión de su evento, bloqueando el progreso hacia los restantes estados. Por fuera de esta circunstancia, el equipo sin embargo continuó la ejecución del proyecto para cumplir con los objetivos académicos del Proyecto Final de Carrera, volviendo al sistema completamente operable y funcional.

Alpha “Oportunidad”

El estado final alcanzado para el alpha Oportunidad fue “**Viable**” (**Viable – 4/6**), alcanzado parcialmente sin abordar el elemento “*Solución rentable*”.

El motivo por el cual no fue posible continuar con la progresión de elementos y de estados para este alpha obedece a que, de manera similar a la evaluación del alpha anterior, los stakeholders decidieron no adquirir el sistema de software para la gestión del evento evento, por lo que el elemento “*Solución rentable*” nunca pudo ser abordado.

Si bien el alpha no ha podido llegar a su estado final, sí ha alcanzado un avanzado dentro de su ciclo. Esto encuentra justificación en que el representante de los stakeholders tenía clara cuál era la oportunidad que se estaba abordando con el sistema para la gestión del ENIEF 2019.

Alpha “Stakeholders”

El estado final alcanzado para el alpha Stakeholders fue “**Satisfechos para despliegue del software**” (**Satisfied for Deployment – 5/6**), con todos sus elementos alcanzados.

Aunque no fuese posible alcanzar el estado final para este alpha debido a que eso implicaría que el sistema se encuentre en uso por los stakeholders, su progreso obedece a la

colaboración desarrollada entre el equipo de desarrollo y el representante de los stakeholders a lo largo de todo el proyecto, demostrándose esta colaboración útil para generar acuerdo respecto de las necesidades a ser abordadas por el sistema y la correcta implementación de la funcionalidad para ello.

Alpha “Equipo”

El estado final alcanzado para el alpha Equipo fue “**Disuelto (Adjourned – 6/6)**”, con todos sus elementos alcanzados.

La capacidad de abordar completamente todos los estados de este alpha se encuentra justificada en la adaptación del equipo a trabajar como en unidad, mantener el cumplimiento de los compromisos detallados en la planificación y continuar con la generación de entregables de valor producto del trabajo hasta la finalización del proyecto.

Debido a que el equipo no contempla el desarrollo de incrementos posteriores ni de mantenimiento posterior para el presente proyecto, se determinó al equipo como disuelto al final de la ejecución del proyecto.

Alpha “Trabajo”

El estado final alcanzado para el alpha Trabajo fue “**Cerrado (Closed – 6/6)**”, con todos sus elementos alcanzados.

Este alpha ha sido completado debido a que el equipo logró el cumplimiento de las tareas dentro del tiempo asignado para la ejecución del proyecto, pudiendo finalizar todo el trabajo planificado y no planificado.

Alpha “Way of Working”

El estado final alcanzado para el alpha Way of Working fue “**Retirado (Retired – 6/6)**”, con todos sus elementos alcanzados.

Esto se ha logrado gracias a la predisposición, coordinación y adaptación del equipo al way of working planteado y la aplicación de las prácticas de desarrollo seleccionadas, procediendo a pulirlo continuamente y adaptándose a utilizar las herramientas la forma más productiva posible.

Evaluación del equipo del resultado de la ejecución del proyecto

En líneas generales, y dejando de lado el hecho de que el sistema de software desarrollado resultó no utilizado para la gestión del ENIEF 2019, el equipo de desarrollo se encontró satisfecho por el resultado alcanzado, habiendo cumplido con las necesidades manifestadas y los requerimientos surgidos.

Considerando el uso del producto, este fue totalmente académico para la realización de este proyecto final de carrera. La razón de esto fue que los *stakeholders* realizaron un concurso para elegir entre varias propuestas, y concluyeron que la mejor alternativa por cuestiones de presupuesto presentado era una aplicación que realizaron los mismos desarrolladores que habían realizado aplicación utilizada para el evento anterior. Luego del concludido el evento, y en base a la experiencia con la aplicación elegida, los *stakeholders* manifestaron desconformidad con su elección y expresaron que iban a tener en cuenta la plataforma Conferentia para su próximo evento.

En cuanto a los tiempos pautados y los planificados, el proyecto se pudo ejecutar medianamente dentro de los mismos, sin presentar grandes desvíos. Esto es debido a la gestión del proyecto, el trabajo y la coordinación alcanzada por el equipo de desarrollo, gracias a la sinergia generada y a la metodología aplicada, y la comunicación entre los mismos y los *stakeholders*.

La evaluación de los riesgos arrojó resultados exitosos para las pruebas del sistema, demostrándose que los mismos estaban contemplados y cubiertos, pero no se ha podido hacer una evaluación de riesgos en el contexto real del evento debido a que, como se comenta anteriormente, no se ha utilizado en el mismo.

En lo referido al aprendizaje del equipo, es de destacar que la curva de aprendizaje de las herramientas utilizadas para la programación del sistema no resultó problemática para llevar adelante la ejecución del proyecto en tiempo y forma. Esto obedece a que uno de los integrantes del equipo, con cinco años de experiencia profesional al momento de la ejecución del proyecto, pudo servir de referencia para superar varias de las dificultades técnicas que se manifestaron durante la ejecución del proyecto.

Finalmente, es de destacar que las herramientas utilizadas por el equipo para la gestión del flujo de trabajo y la gestión del proyecto, los frameworks Kanban y Essence respectivamente, se mostraron eficientes, efectivas y productivas para su propósito. Los integrantes del equipo creen que, para el desarrollo de iniciativas futuras de software, verían viable el uso de estas herramientas, como así también la propuesta de su uso a terceros en otros proyectos académicos.

Conclusión

Aportes del proyecto

Consideramos al desarrollo del presente proyecto como un aporte de interés dentro de las herramientas disponibles para la gestión digital de eventos, brindando diversificación respecto de las opciones de consumo masivo ya disponibles en el mercado.

Debido a la diferenciación y especialización que ofrece el sistema de software desarrollado, es importante aclarar que no consideramos como competencia directa a estas plataformas, debido a las características que nuestro trabajo ofrece, como ser brindar un mejor acceso a los recursos existentes (PDFs de trabajos académicos, información sobre los disertantes, información turística y de transporte, información sobre empresas, entre otros), un cronograma personalizable en función a los intereses de cada uno, posibilidad de tomar asistencia cómodamente, notificaciones personalizadas, entre otras.

Creemos, en base a expectativas futuras generadas, que existe un nicho de mercado disponible no explotado que se beneficiaría del uso de nuestro proyecto para la organización de eventos académicos. Consideramos que además de los beneficios económicos y logísticos que pueden lograrse mediante el uso de la plataforma, también contamos con el aspecto positivo de *despapelizar* la gestión de eventos académicos, brindando una componente ecológica y siguiendo la línea de las instituciones públicas y grandes empresas³⁶³⁷.

Respecto de los contenidos abordados a lo largo de la carrera de Ingeniería en Sistemas de Información, creemos que el presente proyecto es un aporte para un caso de estudio basado en aplicaciones móviles híbridas en una arquitectura cliente-servidor, con un gran énfasis en brindar una experiencia de usuario de rápida comprensión, debido a las características de los eventos a los cuales se destina, en los cuales el uso es de corta duración y la curva de aprendizaje debe ser mínima.

³⁶ Fortín, L. (9 de febrero de 2018). Cómo son los proyectos para “despapelizar” al Gobierno y desarmar la burocracia. El Cronista. Recuperado el 26 de agosto de 2020, de <https://www.cronista.com/especiales/Como-son-los-proyectos-para-despapelizar-al-Gobierno-y-desarmar-la-burocracia-20180209-0006.html>

³⁷ Leibowitch Beker, F. (2013). Introducción a la Despapelización del Estado. Recuperado el 26 de agosto de 2020, de <https://www.palermo.edu/ingenieria/downloads/CyT%204/CYT402.pdf>

Utilización de la aplicación en el evento

La aplicación no tuvo utilización en el evento específico mencionado anteriormente, debido a que, en el proceso de selección de las distintas alternativas, los organizadores optaron por la aplicación que proponían los desarrolladores de la anterior ENIEF; principalmente por cuestiones presupuestarias. Sin embargo, una vez finalizado el evento, los organizadores del ENIEF 2019 nos mencionaron que, al no haber quedado conformes con la solución que adquirieron, había una gran posibilidad de que usen nuestra plataforma para sus futuros eventos, una vez se retome la organización luego de la pandemia de Covid-19.

Debido a lo anterior y a pesar de que la aplicación no se utilizó en el evento, decidimos desarrollarla para poder cumplir con lo requerido en este proyecto final de carrera. Si bien la mayoría de los requerimientos del incremento fueron realizados específicamente para el evento, nos sirvieron para mejorar de gran manera la plataforma Conferentia en el caso de requerir acondicionarla para eventos futuros.

Experiencia del proyecto

Esta sección final incluye un detalle de los puntos más destacables de lo que representó para el equipo el desarrollo y la ejecución del PFC. Se incluye aquí una puesta en común de parte de los integrantes del equipo respecto de lo que les significó a ambos el transitar este proyecto.

Puesta en práctica de un proceso ágil de desarrollo basado en una framework nuevo para el equipo

El hecho de trabajar bajo un proceso ágil basado en Kanban, un framework novel para ambos integrantes del equipo y no abordado directamente en ninguna asignatura de la carrera de Ingeniería en Sistemas de Información representó un desafío. Su puesta en práctica requirió de estudio y una extensa consulta de bibliografía y recursos para poder aplicarlo de forma efectiva, adaptándose a sus principios y prescripciones para generar la metodología de desarrollo utilizada en el proyecto. El equipo consideró que

Utilización del framework Essence y del SEMAT Kernel para la gestión del proyecto

De manera similar a Kanban para la gestión del proceso de desarrollo, el framework Essence para la gestión de proyectos requirió de aprendizaje y de extensiva consulta de bibliografía, tanto en libros como en el estándar de Essence especificado por el Object Management Group. El equipo considera que optar por la utilización de esta herramienta para la gestión del proyecto fue un acierto, permitiendo además evidenciar de manera clara cómo se progresó en el proyecto y cuáles fueron los diferentes impedimentos y problemas que el equipo hubo de abordar durante su ejecución.

Desarrollo de aplicaciones móviles híbridas con Ionic

Si bien ambos integrantes del equipo anteriormente trabajaron en el desarrollo de aplicaciones móviles para la plataforma Android, la programación de aplicaciones móviles híbridas y las tecnologías para desarrollarlas requirieron de adaptación y aprendizaje para su uso y representaron una experiencia novedosa para el equipo.

Habiendo tenido previamente uno de los integrantes familiaridad con el desarrollo web en Angular, el cual es parte integrante fundamental de Ionic, la curva de aprendizaje para este último pudo ser abordada de manera satisfactoria para cumplir con los tiempos requeridos para la ejecución del PFC.

Experiencia del equipo colaborando en el desarrollo de iniciativas previas

Las colaboraciones entre los integrantes del equipo en iniciativas previas facilitaron el abordaje del trabajo necesario para el desarrollo para el presente PFC, evidenciándose esto en la rápida progresión en los estados de los alphas del área Iniciativa dentro del SEMAT Kernel.

Evaluaciones de riesgos detallados en un proyecto de software

El equipo consideró valioso lo experimentado y aprendido en lo referente a cómo abordar la complejidad que implica identificar, priorizar y adoptar estrategias y técnicas que permitan mitigar el riesgo que implica el implementar funcionalidades determinadas en un sistema de software. Si bien estos conceptos fueron abordados en la asignatura Ingeniería de Software, el equipo se encontró ante un contexto más amplio en el cual poder aplicar estos conocimientos de forma más profunda.

Glosario

Ad hoc

Locución latina. Que está hecho especialmente para un fin determinado o pensado para una situación concreta.

Alpha

Un alpha es un elemento esencial del esfuerzo de desarrollo, el cual es relevante para la evaluación de su progreso y salud. Véase Jacobson, I., Lawson H., Pan-Wei N., McMahon P., Goedicke, M. (2019). *The Essentials of Modern Software Engineering: Free the Practices from the Method Prisons!*, p. 89.

Angular

Framework para aplicaciones web. Véase <https://angular.io>.

API RESTful

Interfaz de programación de aplicaciones que sirve como una capa de abstracción o intermediario basado en el patrón REST (representational state transfer, transferencia de estado representacional). Véase Ferreira (2009).

Aplicación móvil híbrida

Aplicaciones móviles diseñadas en un lenguaje de programación web que permite adaptar la vista web a cualquier vista de un dispositivo móvil.

Backend

En la separación de intereses dentro de la arquitectura de una aplicación, se denomina así a la parte dedicada a lidar con la lógica de negocios del sistema y el acceso y persistencia de datos.

Cadencia

Manera regular de ocurrencia de eventos en períodos de tiempo que se repiten. Aplicable a la generación de entregables en un proceso de software.

Chair

Anglicismo. Coordinador de una sesión de una conferencia académica.

Code review

Práctica de desarrollo de software. Revisión de código por una persona diferente a la que lo escribió.

Código QR

Código de barras bidimensional cuadrada que puede almacenar los datos codificados. Véase ISO/IEC 18004.

Conferencia

Disertación o exposición en público sobre un tema o un asunto.

Conferentia

Plataforma desarrollada en este Proyecto Final de Carrera.

Congreso

Reunión, normalmente periódica, en la que, durante uno o varios días, personas de distintos lugares que comparten la misma profesión o actividad presentan conferencias o exposiciones sobre temas relacionados con su trabajo o actividad para intercambiarse informaciones y discutir sobre ellas.

Control de versión

Gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo.

Cordova

Popular entorno de desarrollo de aplicaciones móviles.

CSS3

Lenguaje de diseño gráfico que permite definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Estándar propuesto por IETF (RFC 7993).

Cuello de botella

Fase de la cadena de producción más lenta que otras, que ralentiza el proceso de producción global.

Deploy

Anglicismo. Despliegue de la aplicación.

Depuración

Eliminación de la suciedad, impurezas o sustancias nocivas de una cosa.

Despapelización

Capacidad de digitalizar de manera efectiva, todos aquellos archivos que aún se conservan en formato papel.

Diseño responsive

Filosofía de diseño y desarrollo cuyo objetivo es adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visitarlas.

ENIEF 2019

XXIV Congreso sobre Métodos Numéricos y sus Aplicaciones. Congreso sobre métodos numéricos con fecha en 2019.

Fast track

En un proceso de desarrollo, los ítems en fast track representan al trabajo que debe ser realizado con prioridad.

Feedback

Anglicismo. Retroalimentación.

Flujo de información

Forma en la que la información pasa de un sector a otro dentro de una organización.

Framework

Estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software.

Funcionalidad

Porción de un programa orientado a satisfacer uno o más requerimientos.

Gestión integral

Conjunto de actividades que interrelacionadas y a través de acciones específicas, permiten definir e implementar los lineamientos generales y de operación de la Institución, con el fin de alcanzar los objetivos de acuerdo a estándares adoptados.

Grado de personalización

Grado en que se puede personalizar un producto.

HTML 5

Lenguaje de marcado para la elaboración de páginas web. Estándar propuesto por IETF (RFC 7992).

HTTP

Protocolo de comunicación que permite las transferencias de información en la World Wide Web. Estándar propuesto por IETF (RFC 2616).

IDE

Integrated Development Environment. Entorno de desarrollo integrado.

IETF

Internet Engineering Task Force. Grupo de Trabajo de Ingeniería de Internet.

Implementación

Proceso post-venta de guía de un cliente sobre el uso del software o hardware que el cliente ha adquirido.

Incremento

Resultado de la ejecución de una secuencia de un modelo incremental.

Ionic

SDK (Software Development Kit) completo y de código abierto para el desarrollo de aplicaciones móviles híbridas.

Jasmine

Framework Javascript para la definición de tests usando un lenguaje natural entendible por todo tipo de personas.

Javascript

Lenguaje de programación interpretado, dialecto del estándar ECMAScript.

JSON

Formato de texto sencillo para el intercambio de datos. Estándar propuesto por IETF (RFC 7159).

JWE

JSON Web Encryption. Estándar abierto basado en JSON propuesto por IETF (RFC 7516).

JWS

JSON Web Signature. Estándar abierto basado en JSON propuesto por IETF (RFC 7515).

JWT

JSON Web Token. Estándar abierto basado en JSON propuesto por IETF (RFC 7519) para la creación de tokens de acceso.

Kanban

Marco de gestión ágil que controla de modo armónico el desarrollo de entregables necesarios en la cantidad y tiempo determinados. Véase la sección “Proceso de Desarrollo” del presente documento.

Karma

Test runner para Angular que permite automatizar algunas tareas de los frames de tests, como jasmine.

Lean Thinking

Metodología de desarrollo de negocios que apunta a pensar cómo organizar las actividades humanas para entregar más valor al entorno y a los individuos reduciendo los recursos utilizados y las redundancias. Véase Womack, 1996.

Librería

Conjunto de implementaciones funcionales, codificadas en un lenguaje de programación, que ofrece una interfaz bien definida para la funcionalidad que se invoca.

Look & feel

Apariencia de un software y lo que siente el usuario cuando interactúa con él.

Métrica

Conjunto de regularidades formales y sistemáticas que permiten la comparación de datos cuantitativos de una misma naturaleza.

Microservicio

Unidad funcional concreta e independiente, que trabaja junto a otros para ofrecer la funcionalidad general de una aplicación.

Notificación push

Mensaje que se envía de forma directa desde el servidor a dispositivos móviles y aplicaciones de escritorio, en todo tipo de sistemas operativos.

npm

Node Package Manager.

Ofimática

Aplicación de la informática a las técnicas y trabajos de oficina.

Panelista

Persona que participa en un debate o en una discusión pública sobre un tema determinado en una conferencia académica.

Performance

Desempeño con respecto al rendimiento de una computadora, un dispositivo, un sistema operativo, un programa o una conexión a una red.

PFC

Proyecto final de carrera.

PHP

Lenguaje de programación de uso general que se adapta especialmente al desarrollo web.

Plataforma de software

Sistema que sirve como base para hacer funcionar determinados módulos de hardware o de software con los que es compatible.

Plugin

Aplicación o programa informático que se relaciona con otra para agregarle una función nueva y generalmente muy específica.

Prototipo

Primer ejemplar que se fabrica de un software, el cual sirve para ganar certezas y reducir incertidumbre en requerimientos sobre de la implementación definitiva posterior.

Regresión

Retroceso en el transcurso de un proceso o acción.

Requerimiento

Petición de una cosa que se considera necesaria.

Retroalimentación

Método de control de sistemas en el cual los resultados obtenidos de una tarea o actividad son reintroducidos nuevamente en el sistema con el fin de controlar y optimizar su comportamiento. Véase *feedback loop* en (Meadows, 2008, p.39).

RFC

Request For Comments. Documentación de protocolos y tecnologías de Internet.

SCRUM

Marco de trabajo para desarrollo ágil de software.

SCSS

Metalenguaje de Hojas de Estilo en Cascada (CSS).

Sesión

Fase o acto temporalmente acotados en que se desarrolla una actividad o un proceso.

Simposio

Reunión de especialistas en una materia para tratar y discutir sobre algo concreto relacionado con su especialidad.

SOAP

Simple Object Access Protocol. Protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Estándar propuesto por IETF (RFC 4427).

Stage

Anglicismo. Se denomina etapa de stage a aquella previa a la etapa de producción, donde se realizan las pruebas de integración en un entorno tecnológicamente similar al de producción.

Suite de testing

Conjunto de casos de prueba.

Tech stack

Se denomina tech stack o stack de tecnologías a la combinación de productos de software y lenguajes de programación utilizados para el desarrollo de un proyecto de software, sea éste una aplicación, página web u otro tipo de producto digital.

Teoría de las limitaciones (Theory of constraints)

Teoría desarrollada por Eliyahu M. Goldratt sobre las restricciones de un sistema.

Test runner

Software que sirve para ejecutar casos de pruebas automáticos.

Token

Valor computado resultado de un proceso de tokenización.

Tokenización

Valor computado resultado de un proceso de tokenización.

Trello

Software de administración de proyectos.

TypeScript

Superconjunto de JavaScript, que esencialmente añade tipos estáticos y objetos basados en clases.

Unit testing

Forma de comprobar el correcto funcionamiento de una unidad de código.

UTF-8

Formato de codificación de caracteres Unicode e ISO 10646 que utiliza símbolos de longitud variable. Estándar propuesto por IETF (RFC 3629).

Way of working

Anglicismo. Traducido como "modo de trabajar".

WebStorm

IDE para el desarrollo de software con JavaScript.

Bibliografía y referencias

Libros

Hammerberg, M. & Sundén, J. (2014). Kanban in Action. Manning Publications, 1st edition.

Jacobson, I., Pan-Wei N., McMahon P., Spence I., Lidman, S. (2013). The Essence of Software Engineering: Applying the SEMAT Kernel. Addison-Wesley Professional, 1st edition.

Jacobson, I., Lawson H., Pan-Wei N., McMahon P., Goedicke, M. (2019). The Essentials of Modern Software Engineering: Free the Practices from the Method Prisons!. ACM Books, 1st edition.

Beck, K. M. & Beedle, M. (2001). Manifesto for Agile Software Development. Ward Cunningham.

Sommerville, I. (2010). Software Engineering. Pearson Education, 9th edition.

Brooks F. (1995). The Mythical Man-Month. Addison-Wesley, 2nd edition, Pages 13-26.

Hamilton M. (1986). Zero-Defect Software: the Elusive Goal. IEEE Spectrum, vol. 23, no. 3, Pages 48-53.

Papers

Jacobson, I., Pan-Wei N., McMahon P., Spence I., Lidman, S (2012). The Essence of Software Engineering: The SEMAT Kernel. Communications of the ACM, Vol. 55 No. 12, Pages 42-49.

Artículos

Selmanovic, D. A Beginner's Guide To Managing Software Development with Kanban and Trello. Recuperado el 26 de agosto de 2020, de <https://www.toptal.com/agile/guide-managing-development-kanban-trello>.

Radigan, D. ¿Qué es kanban? Recuperado el 26 de agosto de 2020, de <https://www.atlassian.com/es/agile/kanban>.

Wikipedia, La enciclopedia libre (2020). JSON Web Token. Recuperado el 31 de mayo de 2020, de https://es.wikipedia.org/wiki/JSON_Web_Token.

Wikipedia, La enciclopedia libre (2020). Prueba unitaria. Recuperado el 28 de noviembre de 2020, de https://es.wikipedia.org/wiki/Prueba_unitaria.

Jones, M., Microsoft, Bradley, J., Ping Identity, Sakimura, N. and NRI (2015). JSON Web Token (JWT). Recuperado el 21 de septiembre de 2020, de <https://tools.ietf.org/html/rfc7519>.

Cómo hacer testing de Software Manual. Recuperado el 22 de diciembre de 2020, de <https://www.tecnologias-informacion.com/testingmanual.html>

Álvarez C. (2015). Karma.js, el concepto de Test Runner. Recuperado el 26 de mayo de 2020, de <https://www.genbeta.com/desarrollo/karma-js-el-concepto-de-test-runner>.

Coding Potions (2018). Angular - Cómo hacer testing unitario con Jasmine. Recuperado el 26 de mayo de 2020, de <https://codingpotions.com/angular-testing>.

BBVA API Market (2016). API REST: qué es y cuáles son sus ventajas en el desarrollo de proyectos. Recuperado el 7 de junio de 2020, de <https://www.bbvaapimarket.com/es/mundo-api/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos/>.

Fortín, L. (9 de febrero de 2018). Cómo son los proyectos para “despapelizar” al Gobierno y desarmar la burocracia. El Cronista. Recuperado el 26 de agosto de 2020, de <https://www.cronista.com/especiales/Como-son-los-proyectos-para-despapelizar-al-Gobierno-y-desarmar-la-burocracia-20180209-0006.html>

Leibowitch Beker, F. (2008). Introducción a la Despapelización del Estado. Recuperado el 26 de agosto de 2020, de <https://www.palermo.edu/ingenieria/downloads/CyT%204/CYT402.pdf>

Web.dev (2019). Progressive Web Apps. Recuperado el 29 de septiembre de 2018, de <https://developers.google.com/web/progressive-web-apps/>