

PROYECTO FINAL DE INGENIERÍA

ESPECIFICIDAD DE LOS DESARROLLOS WEB

Maronna, Pablo Andrés – LU 122004

Ingeniería en Informática

Rosales, Leonardo Germán – LU 121925

Ingeniería en Informática

Tutor:

Oliveros, Alejandro. Universidad Argentina de la Empresa

Marzo 08, 2014



UNIVERSIDAD ARGENTINA DE LA EMPRESA
FACULTAD DE INGENIERÍA Y CIENCIAS EXACTAS



Agradecimientos

Al profesor Ricardo Wehbe, por ayudarnos en el comienzo de la elaboración de este trabajo. Por aportar ideas, puntos de vista y material muy valioso para la tesis.

Al profesor Alejandro Oliveros, por ayudarnos generosamente durante todo el trabajo. Por sus nuevos puntos de vista y sus ideas para reformular algunas ideas y enriquecer el trabajo.

A los entrevistados, Ariel Schabas, Pablo Ceballos, Mariano Quintela y Carlos De Meglio por su aporte desinteresado, por su apertura y buena predisposición. Sin sus testimonios no se podría haber realizado el trabajo.



Resumen

Especificidad de los requerimientos web es un trabajo final de ingeniería que hace un análisis de la ingeniería de requerimientos para sistemas web. El trabajo estudia las particularidades de las aplicaciones web y cómo estas impactan en la etapa de ingeniería de requerimientos. Finalmente se realiza una investigación exploratoria cualitativa de la industria de software argentina, y cómo se gestionan los requerimientos en los proyectos de desarrollo de aplicaciones web. El propósito no es obtener resultados de valor estadístico, sino estudiar en profundidad los casos presentados y hacer un análisis de los mismos.

El trabajo comienza con el desarrollo de un marco teórico, en el que se definen algunos conceptos centrales que se utilizan a lo largo del trabajo. Éste continúa con la presentación de particularidades propias de los sistemas web propuestas por varios autores.

Mediante una serie de encuestas se estudia cualitativamente de qué manera las empresas estudiadas realizan la ingeniería de requerimientos para aplicaciones web. Se investiga si hay conocimiento de las propuestas teóricas de autores especializados, se indaga sobre qué herramientas y técnicas utilizan, sobre los criterios con los que definen su modo de trabajo y sobre qué dificultades encuentran al trabajar con aplicaciones web. También se investiga qué inconvenientes se presentan en la ingeniería de requerimientos y qué propuestas de mejora tienen.

El trabajo concluye que hay un conocimiento bastante amplio de la teoría sobre ingeniería de requerimientos web y que se dominan las herramientas y técnicas descriptas. Asimismo que están familiarizados con las particularidades de las aplicaciones web y los desafíos que presentan. No obstante, se observa que ninguno de los entrevistados trabaja utilizando una metodología tal cual es definida por los autores, sino que emplean herramientas que toman de las distintas metodologías y las aplican según sus necesidades. La elección de la forma de trabajo, herramientas y técnicas a utilizar está dada por criterios propios, particularmente buscando flexibilidad (que sea adaptable a proyectos diferentes); velocidad (que demore lo menos posible en recopilar, documentar y validar requerimientos) y facilidad (que no sea complicada en su ejecución y que sea entendible por perfiles técnicos y no técnicos). Por último, se recopilan conceptos clave para la ingeniería de requerimientos que fueron presentados por los entrevistados: como son la comunicación directa entre los participantes, poseer conocimiento del negocio y utilizar definiciones iniciales no exhaustivas de los requerimientos.



Abstract

Web Development Specificity studies the use of Web Application specific methodologies for requirements gathering in Argentinian software companies.

The work consists of an exploratory qualitative research, using surveys to collect information from different software companies, and understand the methods that are used and the criterion applied to select them.

The results show that these methodologies are known, but not widely used. The factors used to select a methodology include, among others: the flexibility it offers, the speed to gather the requirements, and the ease of use. Other important observations are that it is necessary to keep a good and consistent communication, that it is necessary to have a good knowledge of the business, and that the use of requirements that are not initially extensively defined makes the process of requirement engineering more flexible.

Companies tend to favour these factors over rigid and detailed methodologies. In many cases, the reason is that they need to fulfill business needs to have the Web Application working as soon as possible, and to be able to respond faster and with less overhead to the rapidly changing world of requirements of Web Applications.



Contenidos

Agradecimientos.....	2
Resumen	3
Abstract	4
Contenidos.....	5
Introducción.....	6
Marco teórico	8
Qué es una aplicación web	8
Características propias de una aplicación web	8
Qué es un requerimiento.....	17
Ingeniería de requerimientos	18
Particularidades de la ingeniería de requerimientos de aplicaciones web.....	20
Hipótesis.....	28
Encuesta.....	29
Descripción de la encuesta	29
Ejecución de la encuesta.....	30
Resultados	31
Conclusión.....	56
Glosario	59
Bibliografía.....	63
Anexo A: Metodologías para el desarrollo de sistemas web de Escalona	65
Anexo B: Modelo de guía de preguntas utilizada	69
Anexo C: Resumen de resultados de las entrevistas	79



Introducción

Desde el surgimiento de Internet hasta el día de hoy, el ecosistema de la informática ha sufrido numerosos cambios. Sin duda uno de los cambios más importantes ha sido primero el surgimiento y luego el desarrollo de las aplicaciones web.

Resulta difícil encontrar usuarios que nunca hayan utilizado o no estén familiarizados con algún tipo de aplicación web. Del mismo modo, entre los profesionales de sistemas, es poco común encontrar personas que no hayan trabajado en algún proyecto para la web. Incluso observamos como muchas aplicaciones de escritorio han sido adaptadas para funcionar en la web, y como otras ya no están disponibles en su versión de escritorio.

Este surgimiento y desarrollo tan acelerado ha presentado a los profesionales de sistemas diversos desafíos. Fue necesario el surgimiento de nuevos lenguajes de programación, nuevos perfiles profesionales, nuevas herramientas de análisis y diseño de aplicaciones, nuevas técnicas de desarrollo, y tantos otros cambios.

Son abundantes los autores que han escrito sobre desafíos y nuevos enfoques para el desarrollo de aplicaciones web. Sus propuestas discuten numerosas soluciones a cada uno de los desafíos.

La industria de sistemas en la Argentina no es ajena a estas realidades. El crecimiento de la demanda de puestos de trabajo para este tipo de aplicaciones (Ensinck 2013) nos da una idea de que la industria informática en Argentina está cada vez más ocupada en desarrollos para la web. Todas estas empresas se enfrentan con los desafíos propios de este tipo de aplicaciones.

Para este trabajo, nos propusimos hacer un diagnóstico de cómo se realiza la ingeniería de requerimientos de aplicaciones web en Argentina, y ver si guarda alguna relación con lo que proponen los distintos autores que estudian sus particularidades. No se persigue obtener resultados que se puedan atribuir a toda la industria argentina, pero sí analizar los casos presentados y sus peculiaridades.

Antes de comenzar será necesario definir algunos conceptos que se usan repetidamente a lo largo de este documento. En principio, es necesario definir qué se considera una aplicación web, cuáles aplicaciones entran dentro de esta clasificación y cuáles



no.

Asimismo, se enumeran las diferencias que tienen esta clase de sistemas respecto de los demás desde el punto de vista del usuario, pero especialmente desde el punto de vista del desarrollo.

Luego se define a qué se llama requerimiento, cuál es su función en la ingeniería de requerimientos y cómo se clasifican tradicionalmente.

Finalmente, luego de haber establecido el marco teórico anterior, se detalla qué diferencias proponen distintos autores para la ingeniería de requerimientos web. Por ejemplo, distintas formas de clasificar los requerimientos que surgen de necesidades propias de este tipo de sistemas.

Este marco es contrastado con la realidad observada en la industria argentina con el fin de comprobar si mantienen una relación no desde una perspectiva estadística sino cualitativa. Se desea corroborar también si se conocen las propuestas de los autores, cuáles de ellas son aplicadas y los criterios utilizados para seleccionarlas.



Marco teórico

Qué es una aplicación web

Antes de avanzar sobre la ingeniería de requerimientos, es necesario definir a qué se denomina aplicación web en este trabajo.

Se define aplicación web a un sistema de software que provee recursos específicos para la web, como pueden ser contenido o servicios, a través de una red (Internet o Intranet) utilizando tecnologías web y como interfaz de usuario un navegador web.

De esta definición se desprenden varios puntos importantes. El primero es que se está hablando de una aplicación, o sea, de una porción de software que tiene como objetivo cumplir una o más funciones para resolver algún problema. El segundo es que para su construcción se utilizan tecnologías para web, como lenguajes de servidor, lenguajes de cliente, hojas de estilo y protocolos como HTTP, para proveer servicios y contenido a la web. Por último, que es necesario que la aplicación sea accedida utilizando un navegador web.

Esta definición deja fuera tanto al contenido HTML estático como a los Servicios Web (Web Services, en inglés). El HTML estático no puede ser considerado una aplicación web porque no tiene ningún componente de software, sino que es una forma de presentación de datos. Los Servicios Web no pueden considerarse una aplicación web porque el cliente no interactúa con ellos de manera directa y porque la interacción no es a través de un navegador.

A lo largo de este documento se utilizan los términos “aplicación web” y “sistema web” indistintamente.

Características propias de una aplicación web

En el primer capítulo del libro “Web Engineering: the discipline of systematic development of web apps” (Kappel *et al*, 2006) se detallan algunas características propias de los sistemas web, en comparación con otros tipos de sistemas. Estas diferencias están divididas en tres grupos:

- las relacionadas al Producto: Contenido, Hipertexto y Presentación
- las relacionadas al Uso: Contexto Social, Tecnológico y Natural
- las relacionadas al Desarrollo: Equipo de desarrollo, Infraestructura Técnica,



Proceso, Integración y Evolución.

Se han agregado algunas características que surgen de nuestro análisis de otros autores (Pressman, 2006 y Lowe & Henderson-Sellers, 2001), y otras que se unificaron con las propuestas por Kappel. Los puntos que se introducen a continuación pertenecen a Kappel salvo que se indique lo contrario.

Estas diferencias se describen en las secciones que se encuentran a continuación.

1. Relacionadas al producto

1.1 De contenido

Característica multimedia y centrada en el contenido: Los sistemas web giran en torno al contenido que proveen. Éste puede ser entregado en forma de texto, video, gráficos, animaciones u otros. El contenido puede estar publicado, ser generado dinámicamente o ser transmitido en vivo. Estas características traen aparejados desafíos y requerimientos técnicos especiales.

Demandas de calidad: El contenido presentado por la aplicación es cargado o generado automáticamente. Este contenido puede ser actualizado con una cierta frecuencia. Esta frecuencia es un indicador de calidad percibido por el usuario pero también puede estar sujeto a otros parámetros de calidad como exactitud, consistencia y confiabilidad. Estos requerimientos de calidad también deben ser debidamente reconocidos en la captura de requerimientos.

1.2 De hipertexto

No linealidad del contenido: A diferencia del contenido tradicional, el contenido web permite navegabilidad. O sea, el usuario puede saltar dentro contenidos, realizar consultas o búsquedas, aplicar filtros o realizar otras operaciones. El usuario se mueve de manera libre a través de la información de la aplicación dependiendo de sus necesidades e intereses.



Sobrecarga cognitiva y desorientación: En el punto anterior se estableció que la información en los sistemas web no es lineal como la de un documento. Esto puede provocar que el usuario se sienta perdido entre tanta información o se sienta confundido por un exceso de contenido. Los sistemas web deberían contemplar estos problemas y tratarlos a fin de minimizar estos efectos.

1.3 De presentación

Estética: En contraste con las aplicaciones tradicionales, la forma en que se visualizan los datos es un factor importante en el éxito de las aplicaciones web, especialmente por la alta competitividad del mercado. En ocasiones las aplicaciones web están sujetas a modas y tendencias. Un ejemplo de esto son las aplicaciones de e-commerce.

Ser auto explicativo: No es frecuente que los usuarios tengan documentación que explique cómo se usa la aplicación. Tampoco es común que el usuario reciba capacitación para usar el sistema. Por esto, más allá del aspecto estético, es necesario que la interfaz sea intuitiva de modo que el usuario pueda comenzar a interactuar con la misma de manera rápida y efectiva.

2. Relacionadas al uso

2.1 Contexto social

Espontaneidad de los usuarios, carga impredecible y concurrencia (Pressman, 2006): Los usuarios se comportan espontáneamente. Pueden visitar el sitio en cualquier momento, dejar de utilizarlo o comenzar a utilizar uno de la competencia en cualquier momento. Esto quiere decir que la carga de la aplicación es impredecible, un día se puede tener cien usuarios y al otro día miles. De la misma manera, muchos usuarios pueden decidir tener acceso a la aplicación al mismo tiempo.



Heterogeneidad cultural de los usuarios: Si la aplicación corre dentro de una intranet puede tenerse un mayor conocimiento de los usuarios. Si se trata de una aplicación que es accesible de forma libre a través de Internet, es posible encontrar usuarios de diferentes orígenes, formación y culturas. Estas diferencias deberán ser tenidas en consideración para adaptar la aplicación a las distintas necesidades. En muchas ocasiones inclusive será necesario permitir algunas personalizaciones a la aplicación, por ejemplo, soportar múltiples idiomas, formatos de fechas, monedas de pago, etc.

2.2 Contexto tecnológico

Necesidad de calidad de servicio: Los atributos de calidad de servicio, como disponibilidad, ancho de banda, confiabilidad y estabilidad deben ser debidamente considerados a la hora de comenzar el desarrollo de una aplicación web. En una aplicación tradicional, algunas de estas características (como el uso de ancho de banda) pueden ser conocidas de antemano. En el caso de las aplicaciones web pueden resultar más impredecibles, y por esto deben ser contempladas correctamente.

Soporte multiplataforma: Las aplicaciones web pueden ser accedidas desde plataformas muy variadas, distintos navegadores, distintos dispositivos (especialmente móviles en el último tiempo), diferentes especificaciones (tamaño de pantalla, memoria del equipo, etc.) o incluso cada usuario puede configurar su navegador de forma distinta. Lograr una interfaz consistente, unificada y compatible es un desafío. Las imágenes, aplicaciones, el caché de contenido y las sesiones de usuario deben ser tenidos en cuenta y presentan desafíos nuevos respecto de los sistemas tradicionales.

Desempeño (Pressman, 2006): *Un lapso tan corto como cinco segundos se ha constituido en el límite actual para definir si los contenidos son los adecuados respecto de lo que se busca dentro de una página web. A ello se suma una predominante sensación de que en otros lugares podría haber algo mejor para ver o hacer (experiencia que podríamos llamar zapping web) (Camus, 2009).*



Si la aplicación no tiene un buen desempeño, por ejemplo, si demora mucho en cargar, el usuario puede decidir abandonarla y buscar otra similar, o sentirse frustrado al usarla y desarrollar rechazo. En los e-commerce, por ejemplo, el tiempo de carga del sitio tiene un impacto directo sobre el volumen de ventas.

Vínculo estrecho entre modelo de negocios y arquitectura (Lowe & Henderson-Sellers, 2001): Las aplicaciones web utilizan, por lo general, tecnologías específicas que están fuertemente relacionadas. Un sistema web puede tener una estructura que combina una arquitectura de negocios compleja, una arquitectura de información compleja y una arquitectura basada en componentes. Estas estructuras muchas veces implican cambios en el modelo de negocios para que éste pueda ser soportado por las soluciones que provee la tecnología específica.

Arquitecturas abiertas y modulares (Lowe & Henderson-Sellers, 2001): Aunque no es exclusivo de las aplicaciones web, es muy común que éstas sean el resultado de combinar módulos o componentes prearmados. Como resultado de esto los equipos de trabajo suelen necesitar tener habilidades para la integración de componentes.

2.3 Contexto natural

Globalidad: Los sistemas web pueden ser accedidos desde cualquier posición geográfica. Esta propiedad de globalidad o ubicuidad presenta nuevos desafíos. Por ejemplo, contenidos sensibles a la ubicación geográfica o contenidos disponibles en diferentes idiomas. La globalidad también aumenta las demandas de seguridad, las necesidades de monitoreo en las distintas locaciones, y pruebas para simular la distribución geográfica entre otras.

Disponibilidad: Las aplicaciones web entregan contenido inmediatamente con cada acceso. La disponibilidad 24x7, la velocidad de respuesta y la estabilidad, son atributos que la aplicación debe satisfacer para cumplir las expectativas de los usuarios.



3. Relacionadas al desarrollo

3.1 Equipo de desarrollo

Multidisciplinario: Se puede resumir las características particulares de una aplicación web en que *involucran una mezcla entre publicación impresa y desarrollo de software, entre marketing e informática, entre comunicaciones internas y relaciones externas, y entre arte y tecnología* (Powell *et al.*, 1998). Estas características ponen de manifiesto la necesidad de tener equipos multidisciplinarios, el equipo debe tener una amplia variedad de competencias que para poder llevar a cabo el proyecto de desarrollo de una aplicación web.

Promedio de edad bajo: En promedio, los desarrolladores de aplicaciones web son *significativamente más jóvenes* (Kappel *et al.*, 2006). Por lo general estos profesionales suelen estar menos apegados a antiguas convenciones y presentan mayor interés en implementar nuevas tecnologías y herramientas. Como contrapartida, esta misma característica implica también que los integrantes de estos equipos suelen tener menos experiencia en proyectos.

Desarrollo comunitario: El fenómeno de los desarrollos open source ha revolucionado la manera de trabajar en forma comunitaria. Actualmente existe un gran número de proyectos cuyo código fuente es liberado. Entre estos proyectos se puede encontrar algunos muy conocidos y utilizados. Incluso aplicaciones que se han adaptado para muchos usos e industrias distintas.

La publicación del código fuente de un producto a la comunidad permite la inclusión de desarrolladores que no formen parte del equipo de trabajo. También, a la inversa, puede tomarse un proyecto de la comunidad y adaptarlo a las necesidades de un proyecto, creando una aplicación nueva en base al código creado por la comunidad.

El desarrollo comunitario, no obstante, presenta nuevos desafíos. La coordinación del trabajo, la consistencia, el control de calidad del código, incluso problemas legales referidos a la propiedad intelectual.



3.2 Infraestructura técnica

Heterogeneidad: Las aplicaciones web dependen principalmente de dos componentes: servidores y navegadores. Mientras que los servidores suelen ser configurados y operados por un equipo especializado con conocimientos técnicos del tipo de aplicación que se publica, la configuración de los navegadores de los clientes está fuera del control del equipo que desarrolló la aplicación. Los entornos desde donde se accede a la publicación suelen ser completamente heterogéneos. Esta situación se complica más aún si se tiene en cuenta distintos navegadores, distintas versiones, plugins instalados y distintos dispositivos.

Inmediatez (Pressman, 2006): Las necesidades de negocio hacen que sea apremiante poner el producto en el mercado rápidamente. A veces en tan solo días o semanas.

Inmadurez: La inmediatez presentada en el punto anterior, genera que muchas veces los sistemas (o alguna funcionalidad) sean puestos en producción cuando aún están inmaduros (o sea, contienen bugs, están incompletos o aún no fueron debidamente testeados).

Seguridad (Pressman, 2006): Al estar disponible mediante una red, una gran cantidad de usuarios pueden tener acceso a la aplicación. Mucha de la información puede estar pública, ser guardada temporalmente en el navegador, o ser transmitida por redes inseguras. Esto hace que la necesidad de proteger el contenido sensible o confidencial sea una prioridad.

3.3 Proceso

Flexibilidad: En el desarrollo de aplicaciones web no es posible tener un plan de trabajo rígido, sino que es necesario poder adaptarse flexiblemente a cambios en las condiciones del proyecto. La velocidad e impredecibilidad del negocio son incompatibles con metodologías rígidas.

Paralelismo: Dado que los tiempos de desarrollo suelen ser cortos, puede ser necesario trabajar en varias aplicaciones, componentes o funcionalidades en paralelo.



También puede darse que varias etapas del proyecto se den en paralelo, por ejemplo, que las pruebas de una versión se realicen mientras ya se está desarrollando la siguiente versión de la misma.

3.4 Integración

Interna: Es común que las aplicaciones web sean integradas con aplicaciones existentes, por ejemplo con catálogos de productos o sistemas de información de clientes, para tener disponible su contenido (o servicios) en la web.

Externa: Los sistemas web también suelen integrarse con servicios o contenido de aplicaciones web externas o de terceros. Un ejemplo puede ser los medios de pago en los e-commerce.

3.5 Evolución

Cambio continuo: El entorno web, sus tecnologías y sus estándares están en constante cambio. Estos cambios ofrecen nuevas posibilidades y cambian las condiciones tecnológicas. Diferentes condiciones presentan nuevas maneras de resolver problemas, lo cual puede llevar a volver a desarrollar una funcionalidad con alguna tecnología nueva o de una manera nueva. Asimismo, la velocidad del mercado impulsa cambios en los requerimientos que se traducen en cambios en la aplicación.

Presión competitiva: La web es un ambiente de mucha competitividad. Las presiones de mercado y la necesidad de presencia en la web hacen que muchas veces se manejen tiempos cortos para el desarrollo. Un proceso formal y sistematizado de desarrollo puede no resultar la opción que mejor se ajuste a las necesidades competitivas.

Paso acelerado: Las aplicaciones tradicionales siguen un proceso ordenado, las versiones se planifican con antelación y ciclos de vida entre versiones suelen ser largos, desde algunos meses hasta años. Las presiones competitivas de las aplicaciones web llevan a que la



aplicación necesite estar en constante cambio, para esto se suelen utilizar ciclos cortos que muchas veces son de apenas unos pocos días.

Incertidumbre del cliente (Lowe & Henderson-Sellers, 2001): La presión competitiva, el paso acelerado, los cambios tecnológicos, los cambios de contexto, y otros factores del entorno web llevan a que el cliente experimente un alto grado de incertidumbre. El contexto hace que el cliente tenga dificultades para comunicar sus necesidades, para ver si una cierta solución es la correcta para sus necesidades e incluso para reconocer sus necesidades.

Muchas veces los proyectos están guiados por una visión de negocio más que por necesidades de negocio. Como resultado, puede haber una cierta falta de claridad inicial en el proyecto. Los enfoques basados en prototipos suelen ser una buena opción en estos casos.



Qué es un requerimiento

La norma 610.12 de la IEEE define a un requerimiento de la siguiente manera:

1. Una condición o capacidad que necesita el usuario para resolver un problema o conseguir un objetivo determinado.
2. Una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal.
3. Una representación documentada de una condición o capacidad como se definió en 1 ó 2.

Los requerimientos son, entonces, la forma en que se entienden, expresan, recopilan, documentan, validan y verifican las características que un sistema debe cubrir para satisfacer las necesidades por las que se creó.

La primera conclusión que se puede sacar de esta definición, es que la elección de que un sistema sea desarrollado sobre una plataforma web debería responder a una necesidad, por ejemplo, que sea accesible desde cualquier ubicación.

Otra idea que se desprende de esa definición es que la decisión de la plataforma provoca que la aplicación posea características particulares. Como se vio anteriormente, los sistemas web tienen características que les son propias. Estas características, por ser distintas de las de otras aplicaciones, necesitan tratarse de una manera distinta. Si los requerimientos son características, entonces debe haber requerimientos propios de las aplicaciones web, como por ejemplo requerimientos referentes a la navegación, o requerimientos de seguridad especiales.

En resumen, los requerimientos son la manera en la que describimos las características que debe tener una aplicación web para satisfacer las necesidades o deseos de los usuarios. Si son características, y se estableció que los sistemas web tienen características particulares, se puede inferir que tendrán requerimientos particulares. Específicamente, se necesita una nueva clasificación donde “encasillar” estos requerimientos.



Ingeniería de requerimientos

Dentro de todo proyecto hay fundamentalmente tres tareas relacionadas con los requerimientos.

- Captura de requerimientos
- Validación de requerimientos
- Administración de requerimientos

Durante la *captura de requerimientos* se busca especificar qué funcionalidades debe cumplir el sistema para proporcionar un servicio a sus usuarios. Las herramientas y técnicas para realizar esta captura son variadas, y pueden depender del tipo de tecnología a utilizar, el tamaño del proyecto, las metodologías con las que se esté trabajando u otros factores varios. También es necesario durante esta etapa reconocer las restricciones que tendrá el sistema. Estas restricciones pueden ser tanto de negocio como técnicas. Es una práctica común que las restricciones a la operación del sistema también sean capturadas como requerimientos no funcionales. Todos los requerimientos que se hayan identificados deben ser documentados correctamente. Al igual que durante su identificación, hay variadas herramientas y metodologías para poder documentar o plasmar estos requerimientos. A la hora de presentar herramientas (en una sección posterior de este trabajo) se tomará la definición de requerimientos como una etapa separada, tal y como lo hace Escalona (Escalona & Koch, 2002).

La siguiente actividad es la *validación de requerimientos*. Una vez capturados los requerimientos, deben ser validados. Es necesario cerciorarse de que el requerimiento fue capturado (no se pasó por alto), que fue debidamente entendido y documentado. Es común que durante la validación de requerimientos se descubran puntos pasados por alto, que se descomponga algún requerimiento en dos menores, o que se fusionen dos requerimientos en uno solo. Es importante la validación de requerimientos sea vista, no como una corrección de errores realizados en la etapa anterior, sino como una profundización del análisis del problema. Es frecuente que no se valide en profundidad por considerarse que se está dando a entender que la captura no fue realizada correctamente. Como resultado, el problema no es



analizado correctamente, o no se verifica debidamente con el cliente que haya sido comprendido.

Son muy comunes los errores en la ingeniería de requerimientos de un proyecto, y repararlos en etapas posteriores (durante el desarrollo, por ejemplo) resulta en la mayoría de los casos muy costoso (tanto en recursos como en tiempo). Es por esto que resulta necesario que las dos tareas anteriores se realicen debidamente y de forma iterativa, hasta conseguir un consenso o no descubrir ningún cambio nuevo. El análisis debe ser realizado en equipo, de manera cooperativa con el cliente, y sin perder de vista el objetivo del proyecto.

Lo expuesto anteriormente demuestra que se convierte en imprescindible y fundamental para el éxito del proyecto, una correcta *administración de los requerimientos*. La administración debe acompañar a cada una de las etapas anteriores ayudando a priorizar los requerimientos, a ordenarlos, a manejar la comunicación entre los equipos (que suelen incluir a gente con distintas formaciones, intereses y objetivos), a estimar el esfuerzo de realizar cada requerimiento, a administrar los cambios en los requerimientos, etc. La *administración de requerimientos* representa un verdadero desafío en un aspecto técnico, interpersonal y organizativo pero puede influir fuertemente en el éxito o fracaso de un proyecto.

Algunos autores concluyen que en oposición a la especificación de requerimientos insuficiente y a la constante evolución de los requerimientos, *los enfoques y métodos para construir aplicaciones centrados en el usuario tienen un potencial aún sin precedentes para llegar a mejores especificaciones* (Murugesan et al, 2002). El hecho de que las aplicaciones web sean accesibles por muchos usuarios, permiten tener feedback (directo o indirecto) de distintos usuarios en distintos ámbitos. De esta interacción entre el usuario y la aplicación pueden llegar a reconocerse requerimientos en oposición a los métodos tradicionales como son las entrevistas, encuestas y otros. Esta posibilidad es, de todos modos, una cuenta pendiente de la ingeniería de requerimientos. Aún no hay mucho avance sobre la tarea de captura de requerimientos “online”.



Particularidades de la ingeniería de requerimientos de aplicaciones web

Cuando se trabaja en un proyecto de desarrollo de aplicaciones web, no se deberían perder de vista las diferencias respecto de los sistemas tradicionales que se presentaron anteriormente. La ingeniería de requerimientos no es una excepción y es por esto que se necesita del mismo modo un trato distinto, para ajustarnos mejor a las características propias de estos sistemas.

Frecuentemente, al trabajar con distintos tipos de problemas, o distintos tipos de sistemas o incluso distintas metodologías de desarrollo, las propuestas de metodologías incluyen un conjunto de herramientas y tareas con las cuales trabajar. En el caso de las aplicaciones web, y específicamente de la ingeniería de requerimientos, esto no sucede. No nos encontramos con un conjunto de herramientas o tareas específicas que un autor presente para solucionar los problemas inherentes a esta clase de sistemas. En cambio, lo que se pone de manifiesto son diferencias de criterio a la hora de trabajar o puntos a tener en cuenta. Hay temas que son tratados por varios autores, por ejemplo, el componente de navegabilidad de un sistema web es citado comúnmente como un punto a tener en cuenta a la hora de desarrollar una aplicación web.

La primera diferencia sustancial que surge es cómo clasificar los requerimientos. La clasificación tradicional donde los requerimientos son funcionales o no funcionales parecería no ser la más adecuada. Los autores que proponen nuevas categorías en las cuales clasificar los requerimientos.

A continuación se realiza un breve presentación de los nuevos tipos de requerimientos y de las metodologías específicas. Algunas clasificaciones fueron unificadas ya que distintos autores pueden nombrar de distinta manera a clases de requerimientos iguales.

Una primera clasificación posible, más bien simple y tradicional, es la siguiente (Pressman, 2006).

- Requerimientos de **contenido**
- Requerimientos **funcionales**
- Escenarios de **interacción con los usuarios**



Escalona (Escalona & Koch, 2002), por el contrario, hace un trabajo comparativo entre diez metodologías: WSDM, SOHDM, RNA, HFPM, OOHDM, UWE, W2000, UWA, NDT y DDDP. Un breve resumen de cada metodología se encuentra en el Anexo A. De esta comparativa surge un listado de clasificación de requerimientos. En la Tabla I puede observarse cuáles requerimientos utiliza cada metodología.

Las clases de requerimientos que se desprenden de la comparativa son:

- **Requerimientos de datos**, también denominados requerimientos de contenido, requerimientos conceptuales o requerimientos de almacenamiento de información. Estos requerimientos responden a la pregunta de qué información debe almacenar, administrar y entregar el sistema.
- **Requerimientos de interfaz** (al usuario), también llamados en algunas propuestas requerimientos de interacción o de usuario. Responden a la pregunta de cómo va a interactuar el usuario con el sistema.
- **Requerimientos navegacionales**, recogen las necesidades de navegación del usuario.
- **Requerimientos de personalización**, describen cómo debe adaptarse el sistema en función de qué usuario interactúe con él y de la descripción actual de dicho usuario.
- **Requerimientos transaccionales** o funcionales internos, recogen qué debe hacer el sistema de forma interna, sin incluir aspectos de interfaz o interacción. También son conocidos como requerimientos de servicios.
- **Requerimientos no funcionales**, son atributos de calidad o limitaciones que debe tener un sistema. Por ejemplo los requerimientos de portabilidad, de reutilización, de entorno de desarrollo, de usabilidad, de disponibilidad, etc.

Como se ve, La clasificación de Pressman es más reducida que la de Escalona. Por otro lado, es una clasificación más tradicional. No obstante, cabe destacar que el contenido es considerado como un requerimiento en sí mismo. Esto parecería ser una factor común entre la gran mayoría de los autores (ver Tabla I). El contenido de los sistemas web es tan importante como cualquier funcionalidad y necesita una administración especial.

Finalmente, y como en muchas otras propuestas, Pressman define un tipo de requerimientos asociados a la interacción de los usuarios con el sistema.



Otro autor que recomienda una nueva clasificación de requerimientos es Kappel (Kappel 2006). Los tipos de requerimientos que expone son:

- **Requerimientos funcionales:** funcionalidades que el sistema debe ofrecer. Pueden describirse utilizando escenarios u especificaciones.
- **Requerimientos de contenido:** son aquellos contenidos que van a ser presentados por el sistema.
- **Requerimientos de calidad:** son una descripción de la calidad que debe tener el sistema o el nivel de calidad de las funcionalidades o de los servicios brindados por el sistema. Algunos ejemplos son: interoperabilidad, seguridad, confiabilidad, desempeño, tolerancia a errores, usabilidad, facilidad de uso, mantenibilidad, estabilidad, adaptabilidad, portabilidad, etc.
- **Requerimientos de entorno:** son los relacionados con el sistema y el ambiente en el que opera. Describen cómo interactúa con otros sistemas, con el hardware utilizado y con componentes de software anteriores. Al ser una aplicación web, es necesario tener en cuenta que la misma será accesible desde mucho sitios distintos. Esta propiedad de “ubicuidad” presenta algunos desafíos que deberán ser tenidos en cuenta como parte del entorno de operación.
- **Requerimientos de interfaz de usuario:** asociado con la facilidad de uso de la aplicación para el usuario (especialmente porque debe ser entendible en el primer uso sin capacitación previa). Distintos tipos de usuarios interactúan de diferente manera con la aplicación. También incluye todos los puntos asociados a la navegabilidad de la aplicación. Suelen usarse prototipos de interfaz de usuario y navegación para poder validar estos requerimientos.
- **Requerimientos de evolución:** las aplicaciones web suelen ser proyectos en constante evolución. Ciclos cortos de desarrollos e iterativos van agregando nuevas funcionalidades a la aplicación. Estas funcionalidades deben interactuar con todas las demás, e integrarse al sistema sin problemas. Por otro lado, cambios de requerimientos o de negocio pueden hacer que sea necesario redefinir completamente alguna funcionalidad. Además, el crecimiento del producto (por ejemplo, en cantidad de usuarios concurrentes) puede presentar problemas nuevos que necesiten soluciones de



escalabilidad nuevas.

- **Restricciones del proyecto:** son limitaciones asociadas a cuestiones del proyecto. Por ejemplo, limitaciones de orden técnico, regulaciones gubernamentales, estándares obligatorios, presupuesto, tiempo, etc. Estas limitaciones al comportamiento son, en la mayoría de los casos, no negociables. O sea, el cumplimiento de una ley (por ejemplo la factura electrónica) no es optativo y además debe implementarse con ciertas restricciones tanto técnicas como legales que están definidas con antelación.

En la siguiente tabla se resumen las clasificaciones vistas anteriormente:



Tabla I: comparativa de clasificaciones de requerimientos y autores

Tipo de requerimiento	Escalona										Kappel	Pressman	
	W S D M	S O H D M	R N A	H F P M	O H D M	U W E	W 2 0 0 0	U W A	N D T	D D P			
Datos/contenido	X	X	X	X	X	X		X	X	X	X	X	X
Interfaz/Presentación		X	X	X	X	X		X	X	X	X(*)		
Navegacionales			X	X	X	X	X	X	X	X	X(*)		
Personalización	X					X	X	X	X	X			
Transaccionales		X	X				X	X	X	X			
Funcionales											X	X	
No Funcionales	X			X		X			X	X			
Calidad											X		
Entorno											X		
Evolución/Adaptación											X		
Restricciones de proyecto											X		
Interacción con el usuario													X

(*) Kappel unifica los requerimientos de interfaz de usuario y navegabilidad en una sola categoría.

Hasta aquí se exploraron tres clasificaciones de requerimientos distintas (Pressman, Escalona y Kappel). Otras posibles diferencias son presentadas por Lowe (Lowe, D., Henderson-Sellers, B., 2001) y están relacionadas con las tecnologías específicas y las diferencias organizacionales, o sea, cómo usa cada organización los sistemas.



- Diferencias técnicas:
 - Vínculo entre modelo de negocios y arquitectura
 - Arquitecturas abiertas y modulares
 - Tecnologías en constante y rápido cambio
 - Importancia del contenido
 - Mayor énfasis en la interfaz de usuario
 - Mayor importancia en los atributos de calidad (usabilidad, performance, robustez)
- Diferencias organizacionales:
 - Incertidumbre del cliente
 - Cambios en los requerimientos de negocio
 - Tiempo reducido para las entregas iniciales
 - Alta competitividad
 - Evolución y mantenimiento más gradual (fine grained)

Sin entrar en detalle en cada uno de los puntos presentados por Lowe, se puede observar que no es más que un listado de obstáculos a superar. Lowe presenta desafíos inherentes a las características de las aplicaciones web.

Lowe es también el autor original de la elicitación guiada por el diseño (DDDP) y que forma parte del trabajo realizado por Escalona.

La propuesta consiste en realizar la captura, definición y validación de requerimientos durante el proceso de diseño. El proceso se basa en el uso de prototipos para ayudar al cliente en la exploración de las posibles soluciones y de los problemas que tienen que ser resueltos. Los usuarios o clientes definen sus requerimientos basándose en la observación o trabajo con estos prototipos. Es un proceso iterativo que consiste en reducir la incertidumbre del cliente.

La siguiente figura, llamada modelo del reloj de arena del desarrollo de los sistemas web (Lowe & Henderson-Sellers, 2001), representa lo que sucede comúnmente en el desarrollo de aplicaciones web.

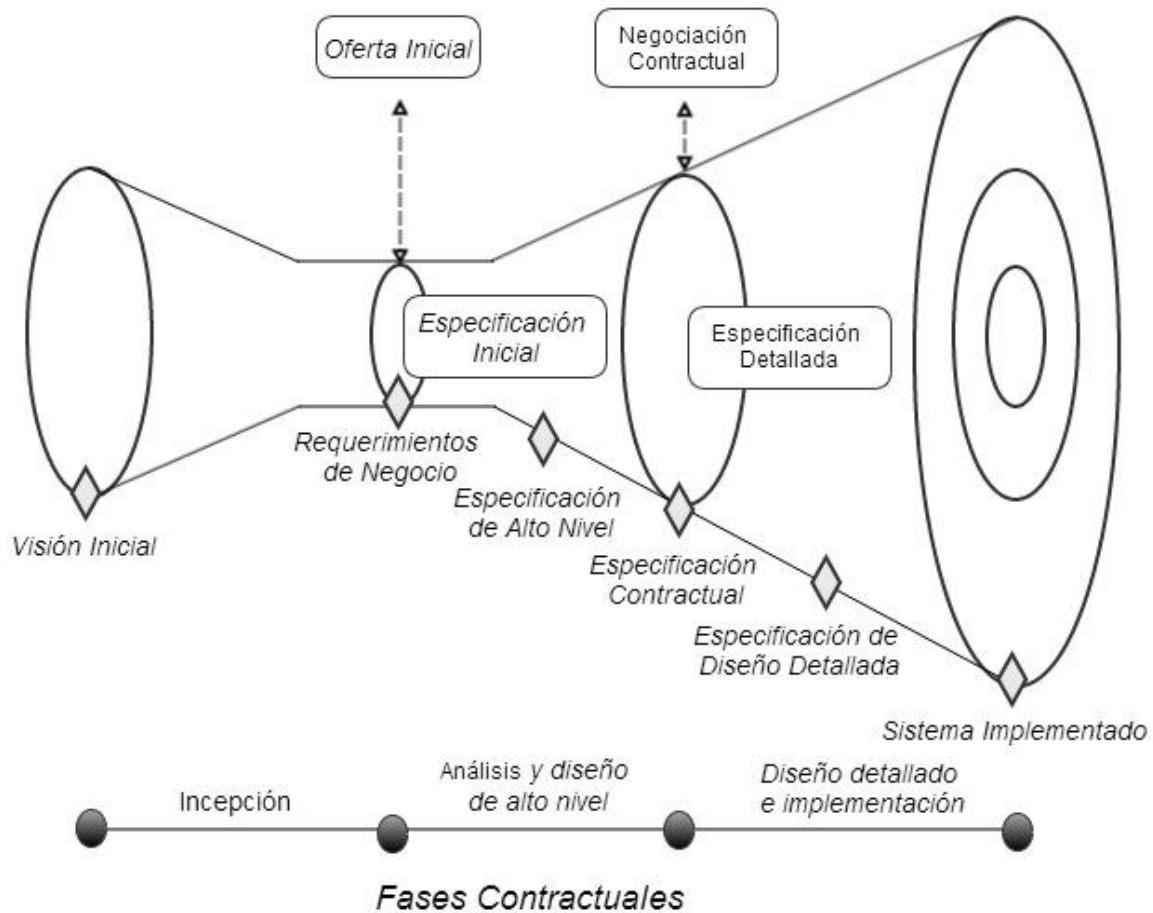


Figura 1: modelo de reloj de arena del desarrollo de aplicaciones web

El conocimiento que se tiene sobre un sistema varía a lo largo del ciclo de vida de desarrollo. El alcance del sistema a desarrollar suele limitarse al momento de elicitar los requerimientos, y desde este momento el conocimiento del sistema aumenta hasta alcanzar su máximo cuando el sistema se encuentra implementado.

Resulta interesante observar que no existe una etapa separada para el diseño. Más bien, esta etapa se encuentra dividida en dos niveles; un diseño de alto nivel concentrado en la arquitectura estructural de la solución, y un diseño detallado concentrado en la arquitectura de los módulos. El primero de estos niveles ha sido incorporado en la elicitación de requerimientos, o fase de análisis. El segundo nivel, de diseño detallado, ha sido incorporado en la fase de producción, o de desarrollo del sistema.



Otro punto interesante de este modelo, es la revalorización de los requerimientos en el ciclo de vida de desarrollo de un sistema. De esta forma, la elicitación no queda como una etapa aislada y restringida, sino que forma parte del diseño y hasta del desarrollo. Más que una etapa, es un proceso que acompaña el desarrollo.



Hipótesis

Lo expuesto hasta aquí consiste en una descripción del estado del arte de las técnicas y metodologías que exponen los diferentes autores para ejecutar el proceso de elicitación de requerimientos para aplicaciones web.

Para este trabajo, se propuso realizar un diagnóstico de la utilización de estas prácticas un conjunto de empresas en Argentina y comprobar si guarda relación con lo que proponen los distintos autores que estudian sus particularidades.

Durante el período de recopilación del material existente en la materia, se descubrieron múltiples metodologías. Si bien entre distintos autores se encontraron puntos en común, la nomenclatura utilizada por cada uno es distinta. Esta falta de consenso nos hace suponer que el trabajo realizado en este campo de investigación no se encuentra lo suficientemente maduro y desarrollado. También, que muchas de las metodologías expuestas se encuentran aún en el ámbito de la investigación, y que por esto mismo su conocimiento general y aplicación es aún limitado. La ausencia de fuentes sobre la materia en la Argentina tratando estos temas es también un factor importante en la adopción de las prácticas expuestas en este país. En base a estos supuestos, se continuó el trabajo bajo la hipótesis de que estas prácticas específicas para las aplicaciones web no son utilizadas, sino que se utilizan las mismas prácticas que para aplicaciones tradicionales.

Al no contar con datos sobre la aplicación de estas prácticas en el mercado argentino, decidimos realizar una investigación exploratoria (y no estadística) y luego indagar sobre estos resultados, con el fin de estudiar las distintas razones por las cuales las distintas prácticas son seleccionadas.

A continuación se contrasta la hipótesis antes expuesta. Para ello se realizó un estudio observacional, con el fin de obtener datos e información sobre cómo realizan la elicitación de requerimientos en empresas argentinas que desarrollan aplicaciones web.



Encuesta

Descripción de la encuesta

Para lograr la corroboración de nuestra hipótesis se realizó un estudio observacional de cuatro empresas argentinas, con el fin de recopilar información, identificar y analizar las particularidades de los desarrollo de aplicaciones web. El método elegido consistió en realizar entrevistas en profundidad a compañías que desarrollan aplicaciones web utilizando una guía de preguntas administradas por los entrevistadores. La guía de preguntas se instrumentó con un cuestionario con preguntas abiertas y cerradas.

El cuestionario ha sido elaborado de manera de poder conocer cómo se trabaja en la compañía entrevistada, y contrastar si se conocen y utilizan los criterios propuestos por los distintos autores.

Cada pregunta del cuestionario posee un apartado para permitir tomar notas sobre datos que no hayan sido considerados en el momento de diseñar el cuestionario. Estas anotaciones son utilizadas para realizar un análisis cualitativo de los resultados, y permitir, de ser necesario, la formulación de nuevas hipótesis.

El cuestionario comienza con cuatro preguntas de datos contextuales, para conocer el perfil del entrevistado, la empresa donde se desempeña y de qué manera se realiza el proceso de elicitación de requerimientos.

La quinta pregunta consiste en una pregunta cerrada, dividida en tres subsecciones; captura de requerimientos, definición de requerimientos y validación de requerimientos, que derivan del análisis realizado de los autores presentados durante el trabajo. El objetivo de esta pregunta es identificar si las técnicas son conocidas por el entrevistado, y cuáles son utilizadas. Se permitió al entrevistado presentar alguna herramienta no incluida en la lista y comentar de qué manera las utilizan.

La sexta pregunta del cuestionario incluye una lista de actividades y herramientas para el tratamiento de requerimientos que presenta Escalona (DFD, diagramas de clases, etc.). Nuevamente, se pregunta si el entrevistado conoce o no la herramienta y si la usa.

La séptima pregunta consiste de una lista de características relacionadas al producto, al uso y al desarrollo de los sistemas web presentadas por Kappel. Se solicitó al entrevistado que



indique, según su criterio, la importancia de cada característica en el momento de elicitación asignándole un puntaje.

Finalmente, se incluyó una serie de preguntas abiertas (preguntas 8, 9 y 10) donde el entrevistado pueda dar su opinión sobre el proceso de captura que utiliza actualmente, y también donde pueda proponer mejoras a la metodología que utiliza junto con algunos comentarios finales.

Al realizar la entrevista se dejó espacio para que el entrevistado pueda pedir aclaraciones sobre alguna de las preguntas o comentarios sobre algunas de sus respuestas. Antes de cada pregunta se presentaba el contexto y una explicación de lo que se buscaba obtener.

La guía de preguntas utilizada para la entrevista se encuentra en el anexo B. En el anexo C se pueden encontrar los resultados de todas las entrevistas.

Ejecución de la encuesta

La población seleccionada para ejecutar la encuesta consiste en cuatro empresas argentinas con distintas características. Las empresas seleccionadas pertenecen a distintos sectores del mercado argentino, y tienen distinto tamaño y actividad principal con el fin de evitar dependencias entre estas variables y el resultado de la encuesta.

Todas las empresas tienen nombres reconocidos en distintos mercados, como son el desarrollo de software, el comercio electrónico, las telecomunicaciones y el trading de commodities. Los tamaños son también muy variados, desde 200 a 5000 empleados, con operaciones en más de un país. En todos los casos, las empresas cuentan con procesos definidos para la elicitación de requerimientos.

Las empresas no fueron seleccionadas al azar, sino que se realizó una selección de entre empresas con las que se pudo establecer contacto y acordar una entrevista. A pesar de que una muestra tan pequeña no es suficiente para presentar conclusiones que puedan tener validez estadística, se buscó variedad tanto en las actividades principales de las empresas, como en el tamaño, modalidades de desarrollo y tecnologías utilizadas.

Algunas de la entrevistas se ejecutaron de forma presencial, y otras vía teleconferencia. Los resultados de cada una se exponen en la siguiente sección.



Resultados

Los cuestionarios fueron completados por los encuestadores, como resultado de una entrevista con el encuestado. Si bien las preguntas estaban predefinidas, se consideraron los comentarios adicionales hechos por los encuestados, y se incorporaron en el cuestionario en los espacios para notas de cada sección.

A continuación, se describe un resumen de los resultados de cada entrevista, y luego se hace un análisis de los resultados.

Los resultados completos obtenidos de las entrevistas se pueden encontrar en el Anexo C de este trabajo.

Perfil de la empresa y el entrevistado

Esta sección resume los resultados de la pregunta 1 en la cual se hace un relevamiento del perfil de la empresa y el entrevistado.

Tabla II: resumen de los entrevistados

Empresa	Actividad principal	Tipo de empresa	Cant. de empleados	Cant. de empleados en sistemas
A	Retailer de indumentaria online	De inversores extranjeros, con presencia en Latinoamérica. Empresa nueva y en gran crecimiento.	300	20
B	Servicios de telecomunicaciones e infraestructura	Multinacional de capital societario	5000	450
C	Trading de commodities	Multinacional	1000	60
D	Desarrollo de software	Capitales privados, ventas en EEUU y desarrollo en Arg.	200	185



Proceso de elicitación actual

Esta sección resume los resultados de la pregunta 2, donde se pide al entrevistado que resuma el proceso de elicitación que usan actualmente.

Entrevistado A

Los clientes internos definen sus requerimientos en prosa o user story, y éstos se cargan en un product backlog. Semanalmente se analizan los pedidos del backlog, se priorizan, se depuran y de ser necesario se solicita más información al usuario.

También se reconocen requerimientos mediante análisis del uso que dan los usuarios a la aplicación (patrones de comportamiento) y mediante monitoreo constante de errores detectados.

Entrevistado B

Existen varios equipos de desarrollo divididos por aplicaciones pero con el mismo proceso de desarrollo.

Los responsables de cada servicio al cliente (telefonía, internet, hosting o networking) presentan su necesidad al jefe de departamento de IT. Luego de que el requerimiento esté aprobado, un analista funcional se reúne con el responsable del servicio para definir el alcance del proyecto. Se definen los casos de uso y se hace una estimación del proyecto.

Entrevistado C

Depende mucho de cuál es la aplicación en cuestión. Dependiendo del proveedor del desarrollo se utilizan distintas herramientas y procesos. Por ejemplo, para los que es SAP se utilizan ciertos templates, más rígidos y se está más limitado por lo que el módulo permite. En otros casos es más flexible e informal.

Entrevistado D

El área de preventas realiza entrevistas por vía telefónica y correo electrónico al cliente, y realiza wireframes de las interfaces de usuario y user stories. Un arquitecto se encarga de analizarlos, refinarlos, validar con el cliente y luego estimarlos.



La definición es extendida antes de comenzar a implementar la historia de usuario entre el equipo de desarrollo y el cliente.

Diferencias entre sistemas web y aplicaciones convencionales

Esta sección resume los resultados de la pregunta 3, donde se pregunta al entrevistado sobre las diferencias entre las aplicaciones web y las aplicaciones convencionales.

Todos los entrevistados reconocen diferencias entre las aplicaciones convencionales y web. La mayoría considera que las principales diferencias están asociadas a la interfaz. Destacan las diferencias de acceso del cliente, y la dificultad para dar ciertas funcionalidades en interfaces web.

Uno de los entrevistados destacó que se le daba mayor importancia a la modularización cuando se trataba de aplicaciones web. La interfaz y la capa de negocio estaban más claramente diferenciadas y se desarrolla de manera más modular.

Otro de los entrevistados hizo hincapié en la naturaleza sin estado del protocolo HTTP y cómo esto condiciona la forma en la que se piensa la interacción entre usuario y sistema, y por ende, su diseño.

Diferencias en el proceso de elicitación

Esta sección resume los resultados de la pregunta 4, donde se pregunta al entrevistado si realiza algunas distinción entre la ingeniería de requerimientos para aplicaciones web y otro tipo de aplicaciones.

Este punto resulta interesante, ya que a pesar de que todos aceptaron que hay diferencias entre las aplicaciones tradicionales y las aplicaciones web, todos manifestaron que no hay ninguna diferencia en cómo realizan la elicitación si se trata de sistemas web o convencionales.

Aunque todos los entrevistados expresaron utilizar adaptaciones de alguna metodología ágil en algún momento de la entrevista, sólo uno lo identificó como una ventaja al trabajar con aplicaciones web.



Otras de las diferencias que se reconocen son:

- La distribución de una nueva funcionalidad se simplifica, ya que el uso del navegador permite actualizar la aplicación sin necesidad de instalar una nueva versión.
- No es necesario instalar la aplicación en cada cliente ni contemplar problemas de sistema operativo o hardware.
- El navegador permite abstraerse de ciertos problemas.



Metodologías de gestión de requerimientos

Esta sección resume los resultados de la pregunta 5, donde se pregunta al entrevistado qué herramientas de captura, definición y validación de requerimientos conoce y utiliza.

En el modelo de guía de preguntas (anexo B) se añadió una breve descripción de cada una de las metodologías.

Para facilitar la lectura se agregó en las tablas una columna de “puntaje” y de “suma de frecuencias”.

El “puntaje” se calcula sumando la frecuencia (nunca=1, casi nunca=2, a veces=3, casi siempre=4 y siempre=5) por la cantidad de personas que respondieron esa frecuencia por cada técnica. La “suma de frecuencias” muestra cuántas veces se asignó esa frecuencia entre todas las técnicas.

Tabla III: resultados técnicas de captura de requerimientos

Técnica	Conoce	Nunca	Casi nunca	A veces	Casi siempre	Siempre	Pje
Entrevistas	A, B, C, D			A	D	C, B	17
Brainstorming	A, B, C, D			A, B, C, D			12
Concept mapping	A, B, D	A, B	D				4
Sketches y Storyboards (wireframes)	A, B, C, D		B	A, C		D	13
Cuestionarios y Checklists	A, B, C, D	A, B	C, D				6
Comparación de terminología	A, B, C, D	A, B, C				D	8
Análisis de otros sistemas	A, B, C, D		C	B, D	A		12
Suma de frecuencias		7	5	9	2	4	



¿Otros? ¿Cuáles?

A: tickets

B: Existen casos en que los requerimientos ya vienen definidos por el área que los solicita en forma de casos de uso.

C: sharepoint o por email

D: por teléfono o email en lenguaje natural

La primera observación es que las herramientas son generalmente conocidas por los entrevistados.

La empresa A es la que mayor cantidad de técnicas utiliza. Principalmente se basan en funcionalidades otros sistemas, siendo que se dedican a construir un producto para el mercado masivo.

Las empresas B y C utilizan principalmente entrevistas, y suelen coincidir también en el resto de las frecuencia de uso.

La empresa D es una software factory, y tiene que tratar con muchos clientes distintos que en general tienen una idea de lo necesitan pero no una definición. Para poder ayudar al cliente a identificar cuáles son sus necesidades, se utilizan sketches y comparación de terminología. De todos modos, no descartan las entrevistas como herramienta sino que las utilizan casi siempre.

Entre las metodologías de capturas de requerimientos, la más comúnmente utilizada es la entrevista. En mucho menor medida se utilizan sketches, brainstormings y análisis de otros sistemas. Concept mapping y cuestionarios/checklist prácticamente no se usan. Es de destacar que la técnica más utilizada es también la más informal.

Las frecuencias de utilización están desbalanceadas hacia el nunca. “Nunca“, “casi nunca“ y “a veces“ son ampliamente las opciones más comunes, por lo que hay herramientas que claramente no se usan. A su vez, en los casos en que se eligió “siempre“ o “casi siempre“, fue sobre distintas herramientas, por lo que parecería que no existe una herramienta que sea usada universalmente siempre.

En la mayoría de las técnicas los entrevistados optaron por “a veces“, por lo que parecería ser las herramientas se usan en combinación.



Tabla IV: resultados técnicas de definición de requerimientos

Técnica	Conoce	Nunca	Casi nunca	A veces	Casi siempre	Siempre	Pje
Lenguaje natural	A, B, C, D	D		B	A, C		12
Ontologías, glosarios, planillas o patrones	A, C		C	A			5
Escenarios	A, B, C, D	B, D			A, C		10
Casos de uso	A, B, C, D		A	B	C, D		13
Historias de usuario	A, B, D	B				A, D	11
Lenguajes formales	A, B, C, D	B	D	A, C			9
Suma de frecuencias		5	3	5	6	2	

¿Otros? ¿Cuáles?

A: wiki

Nuevamente, para las técnicas de definición, las herramientas son conocidas por los entrevistados.

La empresa A es la que mayor variedad de herramientas utiliza, y junto con la empresa D utilizan siempre historias de usuario. Las empresas B y C coinciden en la utilización de lenguaje natural y de casos de uso, y estas suelen ser además las más utilizadas entre todas las empresas.

Se vuelve a dar la situación en que las herramientas más informales (en este caso el lenguaje natural y anteriormente las entrevistas) están entre las más utilizadas.

Para las herramientas de definición de requerimientos no resulta tan claro identificar una herramienta que no se utilice nunca. Las ontologías no son muy conocidas, y los lenguajes formales no son muy utilizados.

La opción “a veces” vuelve a estar entre las más elegidas. Sin embargo, los técnicas están más repartidas; la elección de frecuencias es más pareja. De todos modos, no es posible identificar una herramienta que sea utilizada universalmente siempre.



Tabla V: resultados técnicas de validación de requerimientos

Técnica	Conoce	Nunca	Casi nunca	A veces	Casi siempre	Siempre	Pje
Reviews o walk-throughs	A, B, C, D			C	D	A, B	17
Auditorías	A, C	A, C					2
Matrices de trazabilidad	A	A					1
Prototipos	A, B, C, D		B	A, C, D			10
Suma de frecuencias		3	1	4	1	2	

¿Otros? ¿Cuáles?

Las herramientas de captura son también conocidas, aunque no tan universalmente como en los casos anteriores.

De nuevo, las opciones están repartidas, pero “a veces” y “nunca” están entre las más comunes.

Se realiza una validación mediante reviews, y en mucho menor medida, mediante prototipos.

Matrices de trazabilidad y auditorías, son las herramienta menos conocidas y además, no son utilizadas nunca por quienes las conocen.

En síntesis, las técnicas más utilizadas son también las más informales. La empresa más nueva es la que utiliza mayor cantidad de técnicas distintas. A su vez, las empresas más grandes y con mayor antigüedad parecieran coincidir en utilizar menor cantidad de técnicas, y esto se puede deber a mayor maduración, o bien a que son más conservadoras en cuanto a la elección de técnicas.

Un análisis de las frecuencias nos muestra que “a veces” y “nunca” son las que más se



eligieron, lo nos dice que no hay técnicas que sean definitivas, hay algunas técnicas que claramente no se usan, y que lo común pareciera ser utilizar una combinación de dos o más técnicas en cada fase.



Técnicas o herramientas de gestión de requerimientos

Esta sección resume los resultados de la pregunta 6 donde se pregunta al entrevistado qué técnicas de de tratamiento de requerimientos conoce y utiliza.

Tabla VI: resultados sobre herramientas para gestión requerimientos web

Actividad / Herramienta	Conoce	Utiliza
Diccionario de datos	B, C, D	C, D
Diagrama de contexto	A, B, C, D	A, C
DFD	C, D	C
Diagrama de clases	A, B, C, D	C, B, D
Diseño de pantallas	A, B, C, D	A, B, C, D
Modelo de datos	A, B, C, D	A, B, C, D
NF. Navegación	A, B, C, D	A, B, C, D
NF. Seguridad	A, B, C, D	A, B, C, D
NF. Mantenibilidad	A, B, C, D	A, B, C, D
NF. Accesibilidad	A, B, C, D	B, C, D
NF. Escalabilidad	A, B, C, D	A, B, C, D
Utiliza algún template	A, B, C, D	C
Utiliza herramienta CASE	A, B, C, D	C
Elicitación y el diseño en paralelo	A, B, C, D	A, C, D

La gran mayoría de las técnicas de gestión de requerimientos son conocidas, y muchas de ellas son utilizadas. Aquí no se pudo hallar una tendencia de uso, sino que todas se usan dependiendo de distintas necesidades.

DFD es la herramienta menos conocida, quizás por ser más común entre las aplicaciones de paradigma estructurado. Las herramientas CASE y los templates, a pesar de



ser conocidos, son los menos elegidos por los entrevistados.

Las herramientas y actividades más generalmente usadas son diseño de pantallas, modelo de datos, diagrama de clases y la identificación de requerimientos no funcionales y diseño en paralelo.

El único entrevistado que contesta que conoce y utiliza todas las herramientas es el C. La razón es que el entrevistado C no tiene equipo de desarrollo propio y terceriza, pero realiza la gestión de requerimientos internamente. Al ser el único entrevistado que se dedica exclusivamente a trabajar con requerimientos es entendible que sea el que más herramientas conoce y usa. Otro factor que influye es que la empresa trabaja en muchos proyectos y muy variados. El entrevistado manifestó que usaban gran variedad de herramientas dependiendo del proyecto (ver preguntas 2 y 8).

En el anexo A se puede encontrar un breve resumen de algunas propuestas de metodologías para la ingeniería web (Escalona & Koch, 2002). La metodología que más se ajusta a las usadas por los entrevistados parecería ser Design-Driven Development Process.

Esta metodología es bastante flexible en cuanto a las herramientas y técnicas a utilizar. Propone entrevistas como herramienta de captura de requerimientos, que fue la más popular entre nuestros entrevistados. Para la etapa de definición sugiere utilizar herramientas de diseño, por ejemplo los casos de uso, también el más popular entre nuestros entrevistados. En último lugar, para la validación recomienda utilizar prototipos, que se encuentra también entre los dos más elegidos.

Otra particularidad de la metodología es que identifica varios tipos de requerimientos. En la tabla VI se puede observar que las distintas clases de requerimientos no funcionales son reconocidas y utilizadas casi unánimemente por los entrevistados.

Para reforzar esta idea, vemos que la mayoría de los entrevistados reconocieron realizar el diseño de la aplicación en paralelo con la ingeniería de requerimientos, que es la principal característica de esta metodología.

Darí la impresión de que trabajar de manera iterativa, logrando que la etapa de diseño y de captura de requerimientos se retroalimenten, permite reducir la incertidumbre del cliente y mejorar la calidad de los requerimientos y del diseño. Esto parecería ser una característica que la vuelve interesante para los entrevistados.



Importancia de las características de las aplicaciones web

Esta sección resume los resultados de la pregunta 7 donde se pregunta qué importancia dan a las distintas características de las aplicaciones web a la hora de la elicitación.

Para facilitar la lectura se agregó en las tablas una columna de “puntaje” y de “suma de importancias”.

El “puntaje” se calcula como sumando la importancia (del 1 al 5) por la cantidad de personas que le asignaron esa importancia por cada característica. La “suma de importancia” cuenta cuántas veces se asignó esa importancia entre todas las características.

Tabla VII: resultados caract. de requerimientos de producto e importancia

Subgrupo	Característica	1	2	3	4	5	Pje
Contenido	Caract. multimedia y centrada en el contenido			C, D		A, B	16
	Demandas de calidad		D	B, C		A	13
Hipertexto	No linealidad del contenido		B	C, D		A	13
	Sobrecarga cognitiva y desorientación			B	A, C, D		15
Presentación	Estética				B, C	A, D	18
	Ser autoexplicativa				B, C, D	A	17
Suma de importancia		0	2	7	8	7	

Todas las empresas calificaron entre 3 y 5 en importancia a la gran mayoría de las características propuestos para las aplicaciones web.

Entre las características de producto las más importantes para los entrevistados son la estética y ser autoexplicativas. Es interesante que ambas características pertenecen a la misma categoría o subgrupo: presentación. Esto se confirma en la tabla VI, donde todos afirmaron utilizar el diseño de pantallas como herramienta. También se ve la relación entre las



características de hipertexto y que los entrevistados consideran requerimientos de navegación unánimemente.

En rasgos generales los puntajes son parejos, no hay ninguna que sea muy poco importante.

Tabla VIII: resultados caract. de requerimientos de uso e importancia

Subgrupo	Característica	1	2	3	4	5	Pje.
Contexto social	Espontaneidad de los usuarios, carga impredecible y concurrencia			B, C	A, D		14
	Heterogeneidad cultural de los usuarios		B	C	A, D		13
Contexto tecnológico	Necesidad de calidad de servicio		B		C, D	A	15
	Soporte multiplataforma		B		A, C, D		14
	Desempeño			B, D		A, C	16
	Vínculo estrecho entre modelo de negocios y arquitectura				A, B, C, D		16
	Arquitecturas abiertas y modulares				A, B, D	C	17
Contexto natural	Globalidad			B	A, D	C	16
	Disponibilidad				B, D	A, C	18
Suma de importancia		0	2	6	20	7	

Nuevamente todos los entrevistados asignaron valores altos de importancia a la mayoría de las características.

Dentro de las características de uso, las más importantes para los entrevistados fueron la disponibilidad y las arquitecturas abiertas y modulares. Ambas se confirmarían en la tabla VI donde la totalidad de los entrevistados afirmó reconocer requerimientos de mantenibilidad



y escalabilidad.

A las anteriores características le siguen en puntaje desempeño, vínculo estrecho entre el modelo de negocios y arquitectura, y globalidad. Las tres características son muy distintas en cuanto a la opinión de los entrevistados. Globalidad está muy dispersa, todos coinciden en el vínculo estrecho entre el modelo de negocios y la arquitectura, y en cuanto a la globalidad las opiniones están polarizadas.

En particular, el contexto social parecería ser el grupo de características menos importante. Era esperable que los entrevistados B y C, que desarrollan aplicaciones para uso interno y en un solo país, asignen menos importancia a estas cualidades. Por otro lado, el entrevistado A, que desarrolla un e-commerce y cuyos usuarios le son desconocidos, y el entrevistado D, que desarrolla aplicaciones para todos tipo de clientes, asignan un valor mayor a esta subcategoría.

A pesar de todas las peculiaridades observadas, no hay gran dispersión entre los puntajes de las características. Parecería que de nuevo no hay características que sean de mucho menor importancia que otras.



Tabla IX: Resultados caract. de requerimientos de desarrollo e importancia

Subgrupo	Característica	1	2	3	4	5	Pje.
Equipo de desarrollo	Multidisciplinario			A, C		B, D	16
	Promedio bajo de edad	C	B		A, D		11
	Desarrollo comunitario	C	B	A	D		10
Infraestructura técnica	Heterogeneidad			D	B, C	A	16
	Inmediatez e inmadurez			A, B	C, D		14
	Seguridad				A, B, D	C	17
Proceso	Flexibilidad			B	A, D	C	16
	Paralelismo				A, B, D	C	17
Integración	Interna	A			D	B, C	15
	Externa				B, D	A, C	18
Evolución	Cambio continuo				B	A, C, D	19
	Presión competitiva		C	B	A, D		13
	Paso acelerado		C		B, D	A	15
	Incertidumbre del cliente			B	C, D	A	16
Suma de importancias		3	4	9	25	15	

Nuevamente se repiten tres tendencias: la mayoría de las opciones están entre 3 y 5; no hay características muy poco importantes, y no hay mucha dispersión entre los puntajes.

Dentro de las características de las aplicaciones web respecto del desarrollo, los entrevistados eligieron como las más importantes a la seguridad, el paralelismo, la integración externa y cambio continuo. Las características también coinciden con lo visto en preguntas anteriores. La seguridad se expresó en la identificación de requerimientos de seguridad (tabla VI). El paralelismo en el proceso coincide con el hecho de que la mayoría de los entrevistados realizan la elicitación y el diseño en paralelo (tabla VI). Todos los entrevistados expresan la



importancia de poder aceptar y dar respuesta al cambio continuo (ver respuestas a preguntas 8 y 9).

La integración externa (e interna), o sea, la capacidad de una aplicación para comunicarse con otras, puede explicarse en cada caso según las peculiaridades de cada entrevistado. El entrevistado A que por ser un e-commerce requiere comunicarse con medios de pago, carriers, entes recaudadores, etc. Los entrevistados B y C que desarrollan numerosas aplicaciones internas que necesitan compartir información para poder dar soporte a la operación de las empresas. Por último el entrevistado D que es una factory desarrolla distintos proyectos y debe dar respuesta a las necesidades de integración que le presente su cliente.

Las siguientes características más importantes son equipo de desarrollo multidisciplinario, heterogeneidad de infraestructura técnica, proceso flexible e incertidumbre del cliente. Equipos de desarrollo multidisciplinarios es uno de los principios de las metodologías ágiles, a las que los entrevistados adhieren en mayor o menor medida. En cuanto al proceso flexible, es una de las principales preocupaciones de todos los entrevistados.

Es interesante destacar que en las tres categorías anteriores (tablas de producto, uso y desarrollo) ninguna de las características podría calificarse como “no importante”. En ninguna hay consenso, ya que siempre alguno de los entrevistados le asigna un valor de importancia distinto. Se puede deducir que la lista de características es acertada y se ajusta a la realidad, ya que no sólo la mayoría de las preguntas tiene importancia alta, sino que además no hay ninguna que sea poco importante para todos los entrevistados.

Continuando el análisis, se puede advertir que no hay características que se destaquen por ser muy o muy poco importantes. Más allá de lo estudiado previamente, no hay gran dispersión entre las importancias que asignan los entrevistados.

Las secciones finales de la encuesta consisten en preguntas abierta, donde el entrevistado puede expresar su opinión, y proponer mejoras.

Se encontraron muchos puntos en común entre los entrevistados, los cuales se tratan en los siguientes apartados.



Sobre el proceso actual

Esta sección resume los resultados de la pregunta 8, donde se pregunta al entrevistado cuál es su opinión sobre el proceso de gestión de requerimientos actual.

Tres de los entrevistados manifestaron alguna forma de disconformidad con el proceso actual. Todos los entrevistados tenían correcciones y mejoras para proponer sobre su método de trabajo con requerimientos. Algunas de las propuestas de mejora se analizarán en la pregunta siguiente.

La informalidad durante la ingeniería de requerimientos es otra preocupación expresada por los entrevistados, que además lo reconocieron como un punto negativo. Hubo acuerdo en que algunas herramientas no son utilizadas de manera consistente, y que algunas etapas se pueden saltar en algunos proyectos. También, que es común que se hagan correcciones a los requerimientos una vez comenzada la implementación. Se mencionó también que la informalidad es propia de la maduración de una empresa en etapa de startup.

Dos de los entrevistados comentan que no es posible que exista un proceso único para la ingeniería de requerimientos, ya que no se adaptaría a sus necesidades. Cualquier propuesta resultaría extremadamente genérica o demasiado rígida. En particular, la preocupación de los entrevistados parecería centrarse en la diversidad. Las software factories, o las compañías con muchos proyectos internos, cuentan con gran variedad de tecnologías y productos que condicionan y hasta definen la forma de elicitar. En la mayoría de los casos, el rol del elicitor necesita un conocimiento fuerte del negocio para compensar estas dificultades.

Finalmente, se desprende de las entrevistas que la elicitación es aún una tarea que depende de quién lo realice. Una mayoría de las tareas son subjetivas, y dependen ampliamente de la experiencia de la persona que las realiza. Este punto vuelve sobre la idea anterior, muchas veces es conveniente que el perfil a cargo de la elicitación esté sumergido en las realidades del negocio para sobrellevar ciertas dificultades a la hora de entender, expresar y documentar requerimientos.



Sobre las mejoras al proceso actual

Esta sección resume los resultados de la pregunta 9, en la cual solicita al entrevistado que enumere posibles mejoras al proceso de gestión de requerimientos actual.

Cuando fueron interrogados sobre cuáles mejoras proponían para sus procesos actuales, los entrevistados dieron respuestas dispares entre sí, pero todas se orientaron a la mejora de la gestión de requerimientos y estandarización de procesos.

Las cinco ideas expresadas fueron las siguientes:

- **Gestión de requerimientos**

- Seguimiento: buscar un método de dar visibilidad y trazabilidad a los clientes. Trazabilidad y visibilidad, pero no del desarrollo (que es más común) sino de los requerimientos. Por ejemplo, publicar el backlog al utilizar metodologías ágiles.
- Incorporación de herramientas informáticas: un entrevistado mostró interés por incluir nuevas herramientas que puedan asistir a la adquisición, documentación y validación de requerimientos. En algún momento su empresa utilizó Rational. Para el entrevistado no era necesario algo tan completo o rígido, pero había alternativas para estudiar. Otro de los entrevistados en otra de las preguntas había comunicado que en su compañía utilizaban un issue tracker y wikis para interactuar con los usuarios que solicitan requerimientos.

- **Estandarización de procesos**

- Dar respuesta a la variedad: buscar alguna solución que permita trabajar de manera más estándar entre proyectos muy variados sin caer en una solución “a medida” de cada proyecto.
- Procesos por tipo de aplicaciones: en contraposición con el punto anterior, otro entrevistado propone agrupar sus aplicaciones y armar procesos adaptados a cada tipo que se ajusten mejor a las necesidades específicas.
- Más especificidad: durante una de las entrevistas se expresó la necesidad de, sin perder flexibilidad, ser más rigurosos en las especificaciones. Dependiendo de la naturaleza del proyecto se utilizaban o no ciertas herramientas, o se



realizaban elicitaciones menos rigurosas para luego corregirlas durante el avance del proyecto. El entrevistado opinó que era necesario definir una especificación mínima obligatoria. Por ejemplo, estableciendo una lista de entregables obligatorios para todos los proyectos.

Sobre los comentarios finales

Esta sección resume los resultados de la pregunta 10, donde se propone al entrevistado que exprese cualquier comentario final que pueda tener.

Sólo dos de los entrevistados quisieron agregar comentarios posteriores a la entrevista. Resulta interesante que ambos expresan la misma idea: la necesidad de responder con flexibilidad.

En cada caso por una motivación diferente. En el primer caso, por una necesidad de responder de manera rápida a las necesidades urgentes del negocio. La rigidez volvería más lento el avance. En el segundo caso, como respuesta a la variedad de proyectos y a dar una respuesta que se ajuste mejor.

Conclusión de la entrevista

La empresa A se encuentra en sus primeros años y en gran crecimiento. Por esta razón, se prioriza acompañar al crecimiento de la empresa y dar rápida respuesta a las necesidades de negocio.

Se conocen y utilizan gran parte de los conceptos presentados, pero de forma informal, ya que los procesos utilizados se encuentran en constante adaptación acorde a la realidad del negocio.

La empresa B cuenta con una sección dedicada exclusivamente a aplicaciones web, pero los procesos están definidos de forma genérica, compartiendo los mismos con el desarrollo de aplicaciones como CRM y ERP.

Si bien se utilizan muchos de los conceptos expuestos, se lo hace de manera informal.

La empresa C trabaja con una gran variedad de tipos de proyectos, por lo que se prioriza la flexibilidad y la rapidez. En muchos casos esto quiere decir que no utilizan métodos formales, y prefieren definir los requerimientos en paralelo con la implementación.



La empresa cuenta con muchas herramientas, distintas tecnologías, herramientas legacy y equipos de desarrollo tercerizado, por lo que han logrado definir un proceso que pueda abarcar tanta diversidad.

El proceso cambia dependiendo fundamentalmente de la aplicación en la que estén trabajando, pero también de otros factores.

En la empresa D se conocen las metodologías y técnicas presentadas por los autores. Utilizan la metodología ágil Scrum y prácticas asociadas a la misma, como por ejemplo: uso de historias de usuario, comunicación constante con el cliente y elicitación guiada por el diseño.

Una acotación importante de esta entrevista fue que el proceso puede no ser lo suficientemente completo, pero esto a su vez le otorga mayor flexibilidad para poder trabajar con distintos clientes.

Cabe destacar que todos los entrevistados manifestaron que utilizan procesos que toman elementos de las metodologías ágiles. También daría la impresión de que los entrevistados coinciden con las propuestas de la metodología Design-Driven Development Process. Esta metodología sugiere ejecutar las etapas de diseño e ingeniería de requerimientos en paralelo y mantener una estrecha comunicación con el cliente para reducir los errores y la incertidumbre. Esto último también coincide con lo propuesto por las metodologías ágiles.

En general, las metodologías de las empresas entrevistadas están en sintonía con las metodologías y clasificaciones propias de las aplicaciones web propuestas por los autores. Sin embargo, esto no está formalizado en los procesos existentes, sino que se utilizan de forma flexible y sin seguir la nomenclatura propuesta por los autores. Esta flexibilidad es vista como una ventaja, ya que permite ajustar la ingeniería de requerimientos ante la variedad de proyectos, clientes y tipos de aplicaciones con las que se trabaja.

Parecería ser que un proceso único no es posible, o es muy rígido para adaptarse a todos los casos posibles y continuar siendo efectivo.

Algunos de los criterios que se utilizan para elegir las metodologías utilizadas que se desprenden de las entrevistas son:

- Flexibilidad: que permitan trabajar con distintos tipos de proyectos, distintos clientes, distintas aplicaciones y distintas tecnologías.
- Velocidad: que les permitan realizar la captura, definición y validación de requerimientos lo más rápidamente posible
- Facilidad: tanto para el equipo a cargo de los requerimientos como para los clientes que informan el requerimiento. Por ejemplo, resulta más fácil que un cliente utilice el lenguaje natural a que conozca cómo utilizar un caso de uso o diagrama de clases.

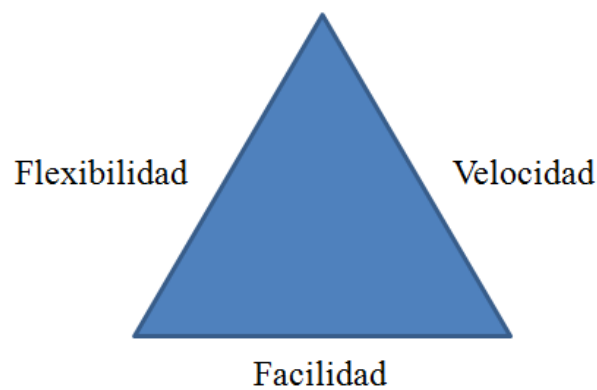


Figura 2: equilibrio de criterios de elección de metodologías

Estas tres características se pueden extraer de las respuestas a las últimas preguntas de la entrevista, donde se cuestiona sobre las opiniones del entrevistado sobre el proceso de gestión de requerimientos actual, posibles mejoras al mismo y los comentarios finales.

A continuación se presentan algunas de las ideas que expresaron los entrevistados, y que nos sugirieron las características mencionadas.

Facilidad

- Entrevistado A: es necesario dar al cliente una herramienta fácil de entender, y que permita dar trazabilidad a los requerimientos.
- Entrevistado B: la captura de requerimientos depende mucho de quien la realice.
- Entrevistado C: los requerimientos para una aplicación web son más difíciles de conseguir y la solución que se le puede brindar es más costosa de lograr. Una reunión o una minuta puede ser suficiente para reconocer un



requerimiento, y esto requiere que la gente que captura los requerimientos tenga mucho conocimiento del negocio.

Flexibilidad

- Entrevistado C: “En nuestro caso, que tenemos muchas aplicaciones y muy variadas, definir un único proceso que se abarcativo a todas las aplicaciones sería buenísimo. Hoy no lo tenemos, pero si lo tuviéramos, y con una cierta flexibilidad, sería buenísimo” “Es muy difícil tener un proceso que abarque toda la diversidad” “Los procesos pueden ayudar, pero hay que tener flexibilidad”.
- Entrevistado D: entiende que el proceso de elicitación debe tener cierta flexibilidad en cuanto al uso de herramientas que se utilizan.

Velocidad

- Entrevistado A: la maduración de la compañía debe ser acompañada, de todos modos, la presión por el cambio y la velocidad del proyecto de desarrollo hace que estos cambios deban realizarse de a poco.
- Entrevistado C: el cliente interno quiere obtener lo que necesita lo antes posible aunque esa respuesta rápida a veces atente contra el resultado. Reconoce como ventaja que un proceso les permita ser más rápidos.

En base a lo observado en las entrevistas se realizó una serie de preguntas adicionales que complementan a las anteriores. También se buscó profundizar sobre las características de velocidad, flexibilidad y facilidad reconocidas anteriormente. Las preguntas son:

- ¿Cuáles creen que son los criterios más importantes al momento de definir la forma en que se van a elicitar los requerimientos?
- ¿Cómo priorizaría entre los siguientes criterios: flexibilidad, velocidad y facilidad?
- ¿Cuáles son los desafíos y problemáticas que encuentran comúnmente al trabajar con aplicaciones web? ¿Estos desafíos se habían identificado como requerimientos? ¿Por qué sí o por qué no?

Las respuestas a estas preguntas se encuentran a continuación.

**Respuestas de la entrevista**

Tabla X: resumen segunda entrevista

Entr.	Ideas principales	Prioridades	Desafíos
A	Velocidad como una forma de dar respuesta a la naturaleza cambiante del negocio. Un requerimiento “acabado y cerrado” demora demasiado tiempo, y podría dejar de ser válido en un contexto de negocio cambiante.	Flexibilidad Velocidad Facilidad	Dificultad en la priorización de requerimientos de disponibilidad y escalabilidad.
B	Comunicación directa como un factor importante para una gestión de requerimientos efectiva.	Flexibilidad Facilidad Velocidad	Integración y seguridad de los datos.
C	Respuesta al negocio de manera veloz y flexible.	Flexibilidad Velocidad Facilidad	Tercerización del desarrollo y comunicación de los requerimientos.
D	Tener conocimiento del contexto ayuda a realizar las preguntas correctas y ayudar al cliente a definir el producto.	Flexibilidad Facilidad Velocidad	Definición de requerimientos de alto nivel muy variada entre proyectos.

Entrevistado A

Define sus prioridades como flexibilidad, luego velocidad y finalmente facilidad. Sus criterios de elección de metodología son estos mismos aunque expresados con otras palabras. Destaca especialmente velocidad como una forma de dar respuesta a la naturaleza cambiante del negocio y los requerimientos. Expresa que conseguir un requerimiento “acabado y cerrado” demoraría demasiado tiempo, y que al momento de cerrarlo podría dejar de ser válido por un cambio en el contexto de negocio. También explica que al ir definiendo de manera más acabada el requerimiento suele crecer el alcance. De esta manera, a la hora de desarrollar el requerimiento es mucho más complejo que al iniciar, lo cual atenta contra el desarrollo.

Sobre los desafíos y problemáticas presenta dos: disponibilidad y escalabilidad. Éstos habían sido identificados como requerimientos, pero no en ciertas etapas del proyecto no



fueron tan prioritarias. La idea que introduce el entrevistado, es que un requerimiento, si no es prioritario difícilmente conseguirá que se le dediquen recursos y tiempo de desarrollo. A veces un requerimiento central, como pueden ser la disponibilidad y la escalabilidad en un e-commerce, no logra que se le asignen recursos hasta que no se vuelve crítico para el negocio.

Entrevistado B

Prioriza la flexibilidad por sobre la facilidad y la velocidad. Destaca también a la comunicación directa como un factor importante para la buena captura de requerimientos.

Los problemas comunes al tipo de aplicación que se desarrolla son la integración con otros sistemas y la seguridad de los datos. Estos requerimientos no suelen ser expresados por el cliente, pero son implícitos y siempre contemplados.

Entrevistado C

Identifica como más importante a la flexibilidad, luego la velocidad y por último la facilidad. Cabe destacar que cuando se le pide que liste sus prioridades las define como primero dar respuesta al negocio, de manera veloz y flexible. En cierta forma invierte entre la primera y segunda respuesta el orden de velocidad y flexibilidad. Esto parecería deberse a que en la primera pregunta, que es más abierta, el entrevistado asoció más estrechamente la velocidad con la posibilidad de dar respuesta al negocio. En la segunda respuesta, con las características presentadas de manera más directa y aislada, prioriza la flexibilidad asociándola con la diversidad de sus proyectos.

Uno de sus desafíos está relacionado con la tercerización del desarrollo. La comunicación es fundamental. Los requerimientos deben ser definidos de manera clara, completa y correcta. Si esto no se realiza correctamente, existe el riesgo de encontrarse con una funcionalidad que no se ajusta a lo que se necesita cuando ya fue desarrollada.

Entrevistado D

El último entrevistado también identifica a la flexibilidad como el criterio más importante al momento de definir la forma en que se elicitan los requerimientos. Luego le sigue la facilidad vista desde el lado del cliente, y por último la velocidad con la que se



ejecuta el proceso.

Un aporte interesante fue el de la definición del contexto antes de realizar la captura de los requerimientos. Esto es importante ya que es una empresa en donde se desarrollan muchos tipos distintos de aplicaciones, y tener conocimiento del contexto ayuda a realizar las preguntas correctas y ayudar al cliente a definir el producto.

Los problemas comunes suelen estar asociados a la variación del alcance debido a una definición de requerimientos inicial no exhaustiva, pero esto mismo aporta ventajas en cuanto a la flexibilidad, por lo que se lo considera también como una ventaja.



Conclusión

La hipótesis inicial de este trabajo fue que no se utilizan las prácticas específicas para las aplicaciones web, sino que en Argentina, los entrevistados utilizan las mismas prácticas para el desarrollo de aplicaciones web que para otro tipo de aplicaciones. Para demostrar nuestra hipótesis, se efectuó un estudio cualitativo, que consistió en realizar una encuesta (entrevistas en profundidad) a un grupo de empresas con una serie de preguntas sobre el conocimiento de las propuestas de los autores, sobre su aplicación y sobre qué opinan de ellas.

El resultado de la encuesta indica que en las empresas entrevistadas no se hace una diferenciación en los procesos en base al tipo de aplicación. Sin embargo, muchas de las herramientas y conceptos presentados por la literatura son conocidos y también utilizados.

Todas las diferencias entre aplicaciones web y tradicionales comentadas por los entrevistados se encuentran entre las diferencias técnicas y organizaciones presentadas por Lowe (Lowe, D., Henderson-Sellers, B., 2001).

Los métodos de captura, definición y validación son conocidos. En su gran mayoría se utilizan con una frecuencia de “a veces” o “nunca”, y son pocos los métodos que se usen siempre. Los preferidos suelen ser aquellos que implican una comunicación directa y en persona, como por ejemplo las entrevistas, brainstorming y review. Estas son a su vez las técnicas más informales.

Mientras más grandes y con mayor antigüedad sea una empresa, pareciera que utilizan menor cantidad de técnicas, y esto se puede deber a mayor maduración, o bien a que son más conservadoras en cuanto a la elección de técnicas.

La mayoría de las respuestas asignan un alto valor de importancia a las características relacionadas al producto, uso y desarrollo de aplicaciones web. No hay ninguna que se destaque por ser muy o muy poco importante, y no hay gran dispersión entre las importancias que asignan los entrevistados. Lo más común pareciera ser utilizar una combinación de varias técnicas.

En los procesos de ingeniería de requerimientos, se otorga mayor importancia a la velocidad de respuesta que a la rigurosidad con las que se aplica una metodología. Es más importante poder responder a nuevos requerimientos velozmente, que ser estricto en el uso de una metodología de requerimientos específica. Esto se debe principalmente a que las

necesidades de negocio requieren que la aplicación esté disponible lo antes posible.

De un análisis de los resultados de la entrevista inicial, surge que las razones para elegir los métodos que se utilizan son variadas, pero tres características parecen ser comunes; flexibilidad para poder aplicarlos en distintos contextos, velocidad para aplicarlos y facilidad de uso.

Al profundizar en este nuevo aspecto, y estudiar los criterios de selección de metodologías, surge que la flexibilidad es el factor que más se prioriza, tanto en la selección de técnicas distintas dependiendo del contexto, como la posibilidad de que permita una fácil actualización del requerimiento. Esto también se cumple en empresas en donde se trabaja en un ambiente cerrado, es decir, mismo cliente y tipo de aplicación.

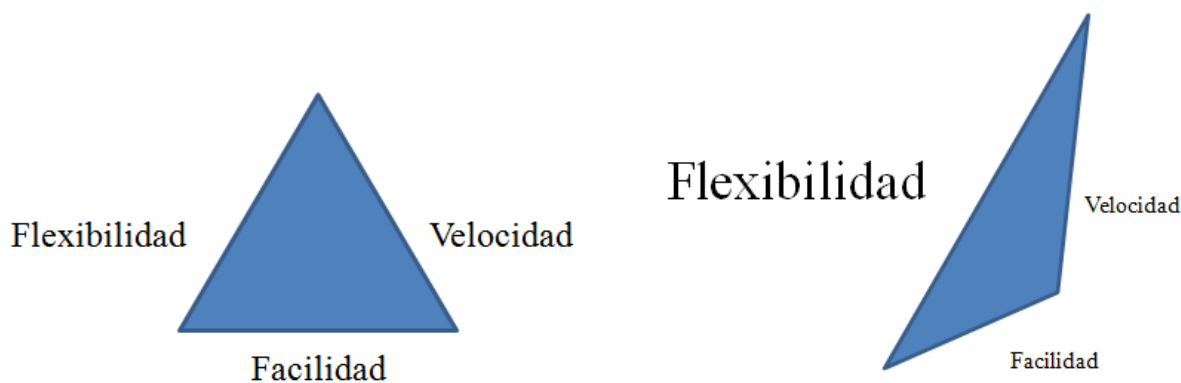


Figura 3: equilibrio vs. proporción de criterios de elección de metodologías

Un requerimiento completo y acabado no siempre es deseable o conveniente. El modelo de reloj de arena del desarrollo de aplicaciones web muestra claramente por qué no es conveniente definir los requerimientos de forma exhaustiva en una etapa inicial. Una definición de requerimiento inicial con mucho nivel de detalle presenta dos problemas que dificultan el posterior trabajo. El primero es que una definición completa requiere de mucho esfuerzo inicial, en el momento del proyecto en que menos conocimiento se tiene de lo que se desea construir. El segundo es que un requerimiento cerrado se vuelve muy rígido y no es fácil de adaptar ante cambios de contexto.

Otras opiniones comunes que se expresan son la importancia de mantener una comunicación constante y fluida, y la importancia de tener conocimiento del negocio antes de comenzar la elicitación.



El desarrollo de aplicaciones web tiene desafíos comunes, y solo aquellos que son recurrentes o técnicos suelen ser contemplados en la etapa de requerimientos, pero los desafíos relacionados con la interacción con el cliente suelen ser más difíciles y dependen mucho del elicitador. Es aquí que la flexibilidad tiene un rol importante para reaccionar ante cambios de contexto imprevistos.

Los entrevistados coinciden en sus preferencias por trabajar de manera iterativa, logrando que las etapas de diseño y de captura de requerimientos se retroalimenten, mientras se obtiene mayor conocimiento de los requerimientos. Las preferencias coinciden también con la metodología Design-Driven Development Process, y a su vez con lo propuesto por las metodologías ágiles.

Entre los entrevistados predomina el uso de metodologías ágiles, y al parecer muchos de los principios de estas metodologías están relacionados con las necesidades de quienes desarrollan aplicaciones web. En particular, la definición de requerimientos en forma escalonada, postergando la definición de bajo nivel, hasta el momento en que el conocimiento es mayor.



Glosario

Adaptabilidad: se refiere a la habilidad para adaptarse. Al hablar de una metodología, se refiere a cuán flexible o adaptable es la misma, para utilizarse en distintas clases de proyectos.

Ágil: se refiere a las metodologías ágiles de desarrollo. Éstas son un grupo de metodologías que proponen métodos, técnicas y herramientas para la ingeniería de software de manera incremental e iterativa. En general proponen trabajar en lapsos cortos de manera colaborativa. La idea general es poder responder más rápidamente a las necesidades del negocio. Entre las más conocidas se encuentran Scrum, Extreme Programming, Kanban y Crystal Clear.

Aplicación cliente-servidor: Es un modelo de aplicación distribuida donde hay un productor de información o servicios llamado servidor, y un consumidor llamado cliente.

Backend: se refiere a las aplicaciones que procesan la información que ingresa a través de un frontend.

Backlog: lista de requerimientos por implementar. En la metodología SCRUM los requerimientos son documentados en un backlog. En cada iteración o “sprint” se priorizan estos requerimientos y se define cuáles se implementarán en esa iteración y cuáles quedarán para una posterior.

Bug: Error de software. Es un error en una aplicación que desencadena resultados no deseados o no esperados.

Caché de contenido: Es un mecanismo mediante el cual se almacena temporalmente el contenido de documentos, sitios, páginas (completos o parciales). El objetivo de estas técnicas es poder entregar más rápidamente el contenido al usuario y bajar la exigencia de los servidores.

Código fuente: Conjunto de instrucciones que ejecuta una computadora para que una aplicación cumpla una o más tareas.

CRM: De la sigla en inglés Customer Relationship Management. Software para la administración de la relación con los clientes.

Desempeño: Cantidad de trabajo realizado por una computadora comparado con el tiempo y los recursos utilizados.

Disponibilidad: Es la capacidad de un sistema de estar en funcionamiento. Generalmente se expresa como porcentaje del tiempo total que debería haber estado en funcionamiento.



También se utiliza como medida de confiabilidad de un sistema.

E-commerce: Comercio electrónico. Se refiere a los sitios que se especializan en la venta de productos online.

Elicitación: Se refiere al proceso de adquisición de conocimiento. En particular, al proceso mediante el equipo sistemas aprende sobre el negocio para poder diseñar un sistema que ayude a que cumplir alguna de las tareas.

ERP: De la sigla en inglés Enterprise Resource Planning. Software para la planificación de recursos empresariales.

Escalabilidad: Se refiere a la capacidad de un sistema adaptarse a una mayor cantidad de trabajo sin perder calidad.

Front End: se refiere a las aplicaciones que interactúan directamente con el usuario.

Funcionalidad: Cualquier acción o tarea que puede realizar un sistema.

Hojas de estilo: Hojas de estilo en cascada. Es un lenguaje que permite describir hojas de estilo que definen el aspecto de páginas web o documentos.

HTTP: es el principal protocolo de comunicación sobre el que se sustenta la web. Una de las características que lo define es que es un protocolo sin estados, o sea, no almacena ninguna información sobre la sesión del usuario ni sus interacciones anteriores.

Informal: Se refiere a aquellas tareas que no son realizadas siguiendo alguna técnica o proceso definido.

Interfaz (de usuario): Se refiere al medio mediante el cual se comunican el usuario y el sistema.

Legacy (sistemas): Sistemas heredados. Son aplicaciones antiguas. Por lo general se utiliza para aplicaciones que por tamaño, importancia, o alguna otra razón no se puede o es difícil reemplazar.

Lenguaje cliente: Se refiere a los lenguajes de programación que son ejecutados en el cliente de una aplicación cliente-servidor. En una aplicación web se refiere a los lenguajes que son ejecutado en el navegador web, como HTML, hojas de estilo y Javascript.

Lenguaje servidor: Se refiere a los lenguajes de programación que son ejecutado en el servidor de una aplicación cliente-servidor. Ejemplos de este tipo de lenguajes puede ser Java, C# o Ruby.



Navegador web: Es una aplicación que permite acceder a sitios web. Inicialmente su principal funcionalidad era mostrar texto, y con la evolución de Internet se fueron incorporando nuevas funcionalidades como la de mostrar contenido multimedia.

Online: Se refiere a los sistemas que están conectados a alguna red. En general describe aplicaciones que están disponibles en alguna red o que consumen servicios de otras aplicaciones en la misma red.

Open source: Se refiere a las aplicaciones cuyo código fuente es publicado para que cualquier otra persona o entidad pueda usarlo libremente, incluso realizarle modificaciones y redistribuirlo.

Performance: (*ver desempeño*)

Plugin: Es una aplicación (por lo general pequeña) que agrega funcionalidad específica a otra (por lo general más grande).

Portabilidad: Es la capacidad de una aplicación de ejecutarse en distintas plataformas o ambientes.

Retailer: distribuidor minorista. Comercio o persona que vende de manera minorista a consumidores finales.

Reutilización (de código fuente): Se refiere a volver a utilizar partes del código fuente de una aplicación o funcionalidad existente para una nueva. Para que la reutilización sea posible es necesario o recomendable que el código tenga una cierta calidad, que respete algunos estándares y que sea desarrollado de manera genérica.

Servicios web: Son métodos de intercomunicación entre aplicaciones o dispositivos. Esta comunicación se utiliza sobre infraestructura web y utilizando estándares definidos por la W3C.

Sesión de usuario: Es la capacidad de una aplicación de mantener la actividad del usuario desde que se conecta a la aplicación hasta su desconexión. En general se lo asocia con un período de tiempo. Actualmente, las aplicaciones tienen la capacidad de mantener la sesión del usuario una vez que se desconecta para que, al volver, pueda continuar en el mismo estado que cuando dejó la aplicación.

Stateless (sin estado): dicho de un protocolo, se refiere a que cada mensaje o solicitud es analizada sin considerar estados o transacciones. Toda la comunicación está dada por



solicitudes y respuestas independientes

Startup: Se refiere a aquellas empresas que están en una etapa incipiente. Generalmente tienen un perfil de riesgo y pueden tener un crecimiento muy pronunciado durante esta primera etapa.

Tercerizar: Se refiere a la subcontratación de servicios.

Usabilidad: Se refiere a la capacidad de un sistema de que el usuario la encuentre fácil de usar. En general se asocia con qué tan intuitiva es su interfaz de usuario, la claridad de su diseño, etc.

Web services: (*ver servicios web*)



Bibliografía

Camus, J. C., Tienes 5 segundos, www.tienes5segundos.cl, 2009

Deshpande, Y., Murugesan S., Ginige A., Hansen S., Schwabe, D., Gaedke, M., White, B., Web Engineering, Journal of Web Engineering, Vol. 1, No.1 (2002) 003-017, Rinton Press, 2002.

Ensinck, M. G., Nuevos oficios digitales, las raras profesiones de hoy, La Nación, 2013, <http://www.lanacion.com.ar/1568028-nuevos-oficios-digitales-las-raras-profesiones-de-hoy> (consulta: 8 de julio de 2013).

Escalona, M. J., Koch, N., Ingeniería de Requisitos en Aplicaciones para la Web, Un estudio comparativo, Universidad de Sevilla, 2002.

Escalona, M. J., Metodologías para el desarrollo de sistemas de información global: análisis comparativo y propuesta, Universidad de Sevilla, 2001.

Kappel, G., Pröll B., Reich S., Retschitzegger W., Web Engineering, The Discipline of Systematic Development of Web Applications, John Wiley & Sons, Ltd., 2006.

Lowe, D., Eklund, J., Development issues in specification of Web systems, University of Technology, Sydney, 2001.

Lowe, D., Henderson-Sellers, B., Characteristics of Web Development Processes, University of Technology, Sydney, 2001.

Lowe, D., Henderson-Sellers, B., Web Development: Addressing Process Differences, University of Technology, Sydney, 2001.



Lowe, D., Haire, B., Henderson-Sellers, B., Supporting web development in the OPEN process: additional tasks, University of Technology, Sydney, 2001.

Murugesan, S., Deshpande, Y., Hansen, S., Ginige, A., Web Engineering: A New Discipline for Development of Web-based Systems, University of Western Sydney Macarthur, 1999.

Murugesan, S., Deshpande, Y., Hansen, S., Ginige, A., Schwabe D., Gaedke, M., White, B., Web Engineering, Journal of Web Engineering, Vol. 1, No.1 (2002) 003-017, Rinton Press, 2002.

Powell, T. A., Jones, D. L., Cutts, D. C., Web Site Engineering: Beyond Web Page Design, Prentice Hall, 1998.

Pressman, R. S., Ingeniería de software, un enfoque práctico, sexta edición, Mc Graw Hill, 2006.



Anexo A: Metodologías para el desarrollo de sistemas web de Escalona

Anteriormente se hizo mención del trabajo realizado por Escalona (Escalona & Koch, 2002). Este trabajo incluye una comparativa de técnicas y modelos de desarrollo para sistemas web.

Los modelos que presenta Escalona son WSDM, SOHDM, RNA, HFPM, OOHDM, UWE, W2000, UWA, NDT y DDDP. Las principales características de cada una pueden resumirse de la siguiente manera.

WSDM: Web Site Design Method. Consiste en detectar los perfiles de usuarios en el entorno del sistema y en base a estos grupos ir modelando el sistema para que responda a las necesidades de cada uno.

- Detección de grupos de usuarios
- Mapa de conceptos de roles y actividades (similar a diagrama de CU)
- Descripción de los grupos de usuarios
- Diccionarios de datos

SOHDM: Scenario-based Object-Oriented Hypermedia Design Methodology. En base a un diagrama de contexto, se identifican los escenarios con los que va a interactuar cada usuario y se utilizan para modelar el sistema.

- Elaboración de un diagrama de contexto (DFD, Yourdon).
- Entidades externas, eventos, escenarios por cada evento
- Diagrama de clases

RNA: Relationship-Navigational Analysis. Es una secuencia de pasos para el desarrollo de aplicaciones web, centrándose fundamentalmente en el flujo de trabajo de análisis.

- Identificación de perfiles
- Identificación de pantallas
- Identificación de objetos, procesos y relaciones
- Navegación



HFPM: Hypermedia Flexible Process Modeling. Propone una larga lista de fases y tareas para obtener los requerimientos.

- Descripción del problema en lenguaje natural
- Casos de uso
- Modelo de datos
- Modelo de clases
- Modelo de interfaz de usuario
- Requerimientos no funcionales (navegación, seguridad, etc.)

OOHDM: Object Oriented Hypermedia Design Model. Consiste en usar User Interaction Diagrams para conocer qué interacciones tendrá el sistema. Es una propuesta para mejorar UML para que se ajuste a aplicaciones web.

- Casos de uso

UWE: UML-Based Web Engineering. Propone 5 tipos de requerimientos y varias técnicas para capturarlos. Relacionados con el contenido, con la estructura, con la presentación, con la adaptación o con los usuarios.

- Casos de uso
- Requerimientos funcionales y no funcionales
- Entrevistas, cuestionarios, checklists, escenarios, glosario
- Validación mediante walk-throughs, auditorías y prototipos

W2000. Se basa en UML y agrega ciertos elementos de multimedia. Separa requerimientos funcionales de navegacionales. Toma los distintos tipos de usuario y funcionalidades de cada rol.

- Requerimientos navegacionales
- Identificación de roles
- Casos de uso
- Caso de uso navegacional



UWA: Ubiquitous Web Applications. Utiliza roles, actividades y relaciones entre usuarios y actividades. Tiene una notación propia.

- Identificación de roles
- Casos de uso
- Utiliza una plantilla
- Utiliza XML para formalizar los requerimientos

NDT: Navigational Development Techniques. Es una técnica para especificar, analizar y diseñar el aspecto de la navegación en aplicaciones web y tiene una herramienta CASE propia.

- Requerimientos de almacenamiento de información
- Requerimientos de actores
- Prototipos de visualización y navegación
- Requerimientos no funcionales
- Herramientas CASE

DDDP: Design-driven Requirements Elicitation. La propuesta consiste en realizar la captura, definición y validación de requerimientos durante el proceso de diseño.

El proceso se basa en el uso de entrevistas para la captura de requerimientos. Para la definición de requerimientos se propone utilizar herramientas de diseño. Finalmente, se utilizan prototipos para ayudar al cliente en la exploración de las posibles soluciones y de los problemas que tienen que ser resueltos. Es decir, los prototipos se utilizan como herramienta de definición y validación de requerimientos. Los usuarios o clientes definen sus requerimientos basándose en la observación o trabajo con estos prototipos. Es un proceso iterativo que consiste en reducir la incertidumbre del cliente. Es necesario trabajar de manera estrecha con el cliente y reforzar la comunicación.

- Requerimientos de contenido
- Requerimientos de interfaces
- Requerimientos de estructura navegacional
- Requerimientos de datos



- Requerimientos de versionamiento
- Requerimientos de seguridad
- Requerimientos de eficiencia
- Requerimientos de adaptación del sistema
- Requerimientos de identificación del usuario
- Utilización de entrevistas



Anexo B: Modelo de guía de preguntas utilizada

Primera entrevista

El objetivo de las preguntas es determinar si las empresas encuestadas utilizan métodos específicos para la definición de aplicaciones Web.

1. Perfil de empresa y entrevistado

Nombre empresa	
Actividad principal	
Tipo de empresa (origen de capital, multinacional/arg)	
Cantidad de empleados	
Cantidad de empleados dedicados a sistemas	
Nombre entrevistado	
Posición del entrevistado	
Tipos de aplicaciones / proyecto con los que trabajan (tamaño, duración, tipos de clientes)	
Tecnologías utilizadas	

2. Describa el proceso de elicitación actual en su empresa

3. ¿Reconocen diferencias entre las aplicaciones web y las aplicaciones convencionales? ¿Cuáles?

4. ¿El proceso de elicitación es distinto cuando se trata de aplicaciones web?



5. Metodologías de gestión de requerimientos

La gestión de requerimientos puede dividirse en tres etapas que consisten en captura de requerimientos, definición de requerimientos y validación de requerimientos. Escalona enumera una serie de métodos que se pueden utilizar en cada una de las etapas. Identificar cuáles se aplican.

Captura de requerimientos

Técnica	Descripción	Conoce	Nunca	Casi nunca	A veces	Casi siempre	Siempre
Entrevistas	Personal o grupal con preguntas abiertas o cerradas.						
Brainstorming	Reuniones grupales informales para generar o compartir ideas.						
Concept mapping	Grafo de conceptos y sus relaciones redactado junto con el usuario para aclarar conceptos del sistema.						
Sketches y Storyboards (wireframes)	Se realizan bosquejos de las interfaces de usuario y con flechas de indica navegabilidad.						
Cuestionarios y Checklists	Mediante cuestionarios se redacta un documento que describa el sistema.						
Comparación de terminología	Definición de terminología específica del						



	cliente y el equipo de desarrollo.						
Análisis de otros sistemas	Utilizar sistemas anteriores como fuente de dónde elicitar procesos y funcionalidades.						
¿Otros? ¿Cuáles?	Se utiliza un sistema de tickets como medio para cargar los issues y un backlog para priorizar.						

Definición de requerimientos

Técnica	Descripción	Conoce	Nunca	Casi nunca	A veces	Casi siempre	Siempre
Lenguaje natural	Redactar los requerimientos utilizando lenguaje natural sin ningún tipo de reglas, templates u otras herramientas.						
Ontologías, glosarios, planillas o patrones	Se utiliza lenguaje informal ordenado por alguna herramienta para poder estructurar la redacción.						
Escenarios	Descripción de las características del sistema a desarrollar mediante una secuencia de pasos utilizando prosa o diagramas.						



Casos de uso	Descripción de los pasos para cumplir una funcionalidad o proceso de un sistema utilizando un template.						
Historias de usuario	Con unas pocas oraciones se describe en lenguaje informal o de negocio qué hace el usuario cuando interactúa con la aplicación para cumplir una cierta tarea.						
Lenguajes formales	Descripción de funcionalidades mediante lenguajes formales, por ejemplo, fórmulas matemáticas.						
¿Otros? ¿Cuáles?	Se utiliza una wiki para poder documentar los requerimientos de manera colaborativa.						



Validación de requerimientos

Técnica	Descripción	Conoce	Nunca	Casi nunca	A veces	Casi siempre	Siempre
Reviews o walk-throughs	Lectura y corrección de la completa documentación o modelado de la definición de requerimientos.						
Auditorías	Revisión de los requerimientos utilizando un checklist predefinido.						
Matrices de trazabilidad	Chequear los objetivos del sistema con los requerimientos identificados						
Prototipos	Aún sin conocer todas las funcionalidades del sistema armar una versión preliminar reducida para que el cliente se haga una idea de cómo será el producto terminado.						
¿Otros? ¿Cuáles?	No						

Notas:



6. Técnicas o herramientas de gestión de requerimientos

Escalona describe una serie de técnicas específicas para el tratamiento de requerimientos de aplicaciones web. Identificar si se utiliza alguna ellas.

Actividad / Herramienta	Conoce	Utiliza
Diccionario de datos		
Diagrama de contexto		
DFD		
Diagrama de clases		
Diseño de pantallas		
Modelo de datos		
NF. Navegación		
NF. Seguridad		
NF. Mantenibilidad		
NF. Accesibilidad		
NF. Escalabilidad		
Templates		
Herramienta CASE		
Elicitación y el diseño en paralelo		

Notas:



7. Importancia de las características de las aplicaciones web

Kappel habla sobre características relacionadas al producto, al uso y al desarrollo propias de los sistemas web.

Indique según su criterio la importancia de cada característica en el momento de elicitación.
(Puntaje de 1 a 5)

Relaciones al producto

De contenido	
Característica multimedia y centrada en el contenido <i>Se tienen en cuenta los generadores de contenido? (estático, dinámico, en vivo)</i> <i>Se tienen en cuenta las diferentes formas de publicación? (texto, animación, video)</i>	
Demandas de calidad <i>Se tiene en cuenta la calidad del contenido? (frecuencia de actualización, exactitud, consistencia, confiabilidad)</i>	
De hipertexto	
No linealidad del contenido <i>Se tienen en cuentas las diferentes formas de navegación? (búsquedas, filtros, vínculos)</i>	
Sobrecarga cognitiva y desorientación <i>Se tiene en cuenta la desorientación que puede causar la no linealidad del contenido? (breadcrumb)</i>	
De presentación	
Estética <i>Se tienen en cuenta la estética como factor de éxito en un medio donde existe una alta competitividad, modas, tendencias?</i>	
Ser autoexplicativa <i>Se tienen en cuenta que el usuario debe poder usar la aplicación sin haber leído antes documentación de la misma?</i>	



Relacionadas al uso

Contexto social	
Esponaneidad de los usuarios, carga impredecible y concurrencia <i>Se considera la carga y simultaneidad de usuarios que va a tener la aplicación?</i>	
Heterogeneidad cultural de los usuarios <i>Se consideran las diferentes culturas y costumbres de los usuarios? (formas de pago, sist medición, idioma)</i>	
Contexto tecnológico	
Necesidad de calidad de servicio <i>Se tiene en cuenta atributos como disponibilidad, uso de ancho de banda, confiabilidad y estabilidad?</i>	
Soporte multiplataforma <i>Se tiene en cuenta el entorno operacional del sistema? (diversidad de navegadores, tamaño de pantallas, distintos dispositivos)</i>	
Desempeño <i>Se tiene en cuenta que el contenido solicitado debe tener un tiempo de respuesta máximo para que el usuario no pierda interés? (5seg, Camus)</i>	
Vínculo estrecho entre modelo de negocios y arquitectura <i>Se tiene en cuenta el modelo de negocios en el momento de proponer una arquitectura?</i>	
Arquitecturas abiertas y modulares <i>Se utilizan módulos o componente prearmados?</i>	
Contexto natural	
Globalidad <i>Se tiene en cuenta la posición geográfica de los usuarios? (huso horario, idioma)</i>	
Disponibilidad <i>Se tienen en cuenta la disponibilidad que debe tener la aplicación?</i>	



Relacionadas al desarrollo

Equipo de desarrollo	
Multidisciplinario <i>Desarrollo, funcional, marketing, diseño</i>	
Promedio de edad bajo <i>Menor experiencia en proyectos y mayor interés en nuevas tecnologías</i>	
Desarrollo comunitario <i>Desarrolladores que no forman parte del equipo, distribuido</i>	
Infraestructura técnica	
Heterogeneidad <i>Infraestructura y desarrollo</i>	
Inmediatez e Inmadurez <i>TTM corto hace que el producto se publique inmaduro</i>	
Seguridad <i>Red pública, almacenamiento en navegador, protección del contenido sensible</i>	
Proceso	
Flexibilidad <i>Adaptabilidad a cambios en las condiciones del proyecto</i>	
Paralelismo <i>Desarrollo o pruebas de componentes en paralelo cuando los tiempos son cortos</i>	
Integración	
Interna <i>Aplicaciones legacy</i>	
Externa <i>Servicios de terceros</i>	



Evolución	
Cambio continuo <i>Cambios tecnológicos o funcionales durante el desarrollo de un producto</i>	
Presión competitiva <i>TTM corto, procesos formales no sirven</i>	
Paso acelerado <i>Ciclos cortos, sin versiones predefinidas</i>	
Incertidumbre del cliente <i>Entorno de aplicaciones web hace que el cliente no pueda comunicar efectivamente los requerimientos. Requerimientos guiados por visión más que por necesidad de negocio.</i>	

Notas:

8. ¿Cuál es su opinión sobre el proceso de gestión de requerimientos actual?

9. ¿Se le ocurre alguna mejora al proceso actual?

10. Otros comentarios

Notas para el entrevistador:

- Se desea saber qué es lo que piensa el entrevistado sobre su forma de trabajar, que quieren modificar, etc. Esto es más interesante que la foto en sí que se captura con una encuesta con pregunta predefinidas.
- Cada pregunta debe tener su contexto y una explicación de lo que se quiere obtener.
- Dar lugar para que desarrolle, explique. La estructura propuesta debe servir de guía, pero si el entrevistado decide extenderse sobre un tema, anotar sus comentarios.



Anexo C: Resumen de resultados de las entrevistas

Primera entrevista

1. Perfil de empresa y entrevistado

A continuación se detallan los datos de los entrevistados. Por cuestiones de confidencialidad, no se indica con qué letra se representa a cada entrevistado.

Actividad principal	Retailer de indumentaria online.
Tipo de empresa (origen de capital, multinacional/arg)	De inversores extranjeros. Con presencia en Brasil, México, Argentina, Chile y Colombia. Las oficinas de Argentina dan soporte a Argentina, Chile y Colombia.
Cantidad de empleados	Cerca de 300. Aproximadamente 100 empleados por país.
Cantidad de empleados dedicados a sistemas	Cerca de 20 personas.
Posición del entrevistado	Gerente de sistemas
Tipos de aplicaciones / proyecto con los que trabajan (tamaño, duración, tipos de clientes)	Proyecto único y continuo. Frontend para clientes (tienda) y backend administrativo.
Tecnologías utilizadas	En general tecnologías abiertas. Linux, Nginx, PHP y MySQL como plataforma. Además memcached, solr, etc.



Actividad principal	Servicios de Telecomunicaciones e Infraestructura
Tipo de empresa	Multinacional de capital societario (NASDAQ:GLBC)
Cantidad de empleados	5000
Cantidad de empleados dedicados a sistemas	450
Posición del entrevistado	Especialista en Aplicaciones Web de Servicios al Cliente
Tipos de aplicaciones / proyecto con los que trabajan (tamaño, duración, tipos de clientes)	Aplicaciones de uso interno, proyectos de integración de aplicaciones, portales de acceso de clientes, portales de gestión de recursos. Aplicaciones Web. Duración de 3 a 6 meses.
Tecnologías utilizadas	SAP, Siebel, ASP.Net, Oracle, SQL Server.

Actividad principal	Trading de commodities
Tipo de empresa (origen de capital, multinacional/arg)	Multinacional
Cantidad de empleados	1000
Cantidad de empleados dedicados a sistemas	30 de planta permanente y 30 tercerizados
Posición del entrevistado	Analista funcional
Tipos de aplicaciones / proyecto con los que trabajan (tamaño, duración, tipos de clientes)	Muy variado. Desarrollos custom, productos enlatados, soluciones adaptadas para ellos. Como ERP, por ejemplo se utiliza SAP. Hay aplicaciones compartidas con otros países y propias. Hay aplicaciones legacy bastante antiguas, aplicaciones de escritorio, aplicaciones web, etc. A modo de comentario, no poseen equipo de desarrollo.
Tecnologías utilizadas	También muy variado. En particular, para los sistemas web, lo más común es .NET sobre Oracle. Para aplicaciones viejas, Natural y Dabas.



Actividad principal	Desarrollo de software
Tipo de empresa	Capitales privados, fuerza de ventas en EEUU y desarrollo en Argentina
Cantidad de empleados	200
Cantidad de empleados dedicados a sistemas	185
Posición del entrevistado	Project Manager
Tipos de aplicaciones / proyecto con los que trabajan (tamaño, duración, tipos de clientes)	Desarrollos a medida a empresas de Estados Unidos. Aplicaciones Web, de escritorio, móviles, aplicaciones empresariales. Duración de un mes a varios años. Los clientes son principalmente startups.
Tecnologías utilizadas	Amplio espectro de tecnologías, entre las que se encuentran Java, .Net, Ruby, iOS y Android

2. Describa el proceso de elicitación actual en su empresa

Entrevistado A

Los clientes internos cargan sus requerimientos directamente en un sistema de tickets. Lo carga en forma de prosa o user story.

Estos pedidos se cargan en un product backlog. Semanalmente se analizan los pedidos del backlog, se priorizan, se depuran y de ser necesario se solicita más información al usuario.

También se hacen algunas tareas de análisis del uso que dan usuarios externos a la aplicación y de estos comportamientos o patrones se generan requerimientos.

Finalmente, hay un monitoreo constante de la aplicación. Cualquier error detectado genera un requerimiento de corrección.

Entrevistado B

Existen varios equipos de desarrollo, divididos por aplicaciones. Las divisiones son ERP, CRM, RRHH y servicios al cliente. Esta última división se encarga principalmente de aplicaciones web de acceso de clientes.



Existe una definición de procesos para todas las divisiones de sistemas, pero en mi caso voy a contestar con respecto a la división a la que pertenezco, que es servicios al cliente.

Los responsables de cada servicio al cliente (telefonía, internet, hosting o networking) presentan su necesidad al jefe de departamento de IT. Luego de que el requerimiento esté aprobado, un analista funcional se reúne con el responsable del servicio para definir el alcance del proyecto. Se definen los casos de uso y se hace una estimación del proyecto.

Entrevistado C

Depende mucho de cuál es la aplicación en cuestión. Dependiendo del proveedor del desarrollo se utilizan distintas herramientas y procesos. Por ejemplo, para los que es SAP se utilizan ciertos templates, más rígidos y se está más limitado por lo que el módulo permite. En otros casos es más flexible e informal.

Entrevistado D

El área de preventas realiza entrevistas por vía telefónica y correo electrónico al cliente, y realiza wireframes de las interfaces de usuario y user stories.

Estos documentos son entregados a un arquitecto, quien se encarga de analizarlos, refinar las user stories, desambiguar con el cliente de ser necesario y luego estimarlos.

La definición es extendida antes de comenzar a implementar la historia de usuario entre el equipo de desarrollo y el cliente.

Nota del entrevistador: todos los entrevistados manifestaron utilizar en mayor o menor medida metodologías ágiles. En ningún caso adhieren completamente a una metodología específica, sino que trabajan de manera alineada a las propuestas del desarrollo ágil.



3. ¿Reconocen diferencias entre las aplicaciones web y las aplicaciones convencionales? ¿Cuáles?

Entrevistado A

En general las aplicaciones web son sin estado (*stateless*, ver glosario) o tienden a eso. Esto está relacionado a la tecnología, al protocolo HTTP. Las aplicaciones se piensan como acciones (o requests) únicas del usuario. Cada acción está conectada a otra siguiente pero todas sin estado. A lo sumo, vía persistencia se puede mantener alguna sesión, pero no dentro de la aplicación misma.

Entrevistado B

Existen diferencias técnicas, pero en particular en la empresa, las aplicaciones web suelen utilizarse para acceso de cliente, y las aplicaciones de escritorio son de uso interno. Las aplicaciones de uso interno suelen tener también acceso vía web, pero la mayor funcionalidad está expuesta a través de aplicaciones de escritorio.

Entrevistado C

Primeramente, reconocen que algunos requerimientos, en las aplicaciones web son más difíciles de adquirir. Por ejemplo, las aplicaciones de escritorio son mucho más ricas en lo que hace a la presentación, a la interfaz. Los requerimientos para una aplicación web son más difíciles de conseguir y la solución que se le puede brindar es más costoso de lograr.

Entrevistado D

Si. Las aplicaciones convencionales suelen ser aplicaciones monolíticas, con una interfaz de usuario propia. Las aplicaciones web, en cambio, son aplicaciones cliente servidor, donde la interfaz de usuario es un navegador web que interpreta código HTML enviado desde un servidor hasta el navegador del usuario que está usando la aplicación.



4. ¿El proceso de elicitación es distinto cuando se trata de aplicaciones web?

Entrevistado A

Al trabajar con proyectos web se trabaja de una manera más ágil. La universalidad de los navegadores permiten que una vez se termina de implementar una funcionalidad pueda estar disponible para todos los usuarios. No es necesario instalar actualizaciones a los usuarios, ni probarlo en distintos sistemas operativos, plataformas, hardware con el que cuenta el usuario. Si bien los navegadores tienen sus diferencias, el impacto es de desplegar una nueva funcionalidad es mucho menor. Esto permite trabajar de una manera más ágil y esto nos lleva a elegir una metodología que se asemeja a las propuestas por las metodologías ágiles.

Entrevistado B

No existe tal separación en la empresa.

Entrevistado C

No diferencian por sistemas web o “no web” sino, como se explicó anteriormente, por la aplicación.

Entrevistado D

No



5. Metodologías de gestión de requerimientos

Captura de requerimientos

Técnica	Descripción	Conoce	Nunca	Casi nunca	A veces	Casi siempre	Siempre
Entrevistas	Personal o grupal con preguntas abiertas o cerradas.	A, B, C, D			A	D	C, B
Brainstorming	Reuniones grupales informales para generar o compartir ideas.	A, B, C, D			A, B, C, D		
Concept mapping	Grafo de conceptos y sus relaciones redactado junto con el usuario para aclarar conceptos del sistema.	A, B, D	A, B	D			
Sketches y Storyboards (wireframes)	Se realizan bosquejos de las interfaces de usuario y con flechas de indica navegabilidad.	A, B, C, D		B	A, C		D
Cuestionarios y Checklists	Mediante cuestionarios se redacta un documento que describa el sistema.	A, B, C, D	A, B	C, D			
Comparación de terminología	Definición de terminología específica del cliente y el equipo de desarrollo.	A, B, C, D	A, B, C				D
Análisis de	Utilizar sistemas	A, B,		C	B, D	A	



otros sistemas	anteriores como fuente de dónde elicitar procesos y funcionalidades.	C, D					
----------------	--	------	--	--	--	--	--

¿Otros? ¿Cuáles?

A: tickets

C: sharepoint o por email

D: por teléfono o email en lenguaje natural

B: Existen casos en que los requerimientos ya vienen definidos por el área que los solicita en forma de casos de uso.

**Definición de requerimientos**

Técnica	Descripción	Conoce	Nunca	Casi nunca	A veces	Casi siempre	Siempre
Lenguaje natural	Redactar los requerimientos utilizando lenguaje natural sin ningún tipo de reglas, templates u otras herramientas.	A, B, C, D	D		B	A, C	
Ontologías, glosarios, planillas o patrones	Se utiliza lenguaje informal ordenado por alguna herramienta para poder estructurar la redacción.	A, C		C	A		
Escenarios	Descripción de las características del sistema a desarrollar mediante una secuencia de pasos utilizando prosa o diagramas.	A, B, C, D	B, D			A, C	
Casos de uso	Descripción de los pasos para cumplir una funcionalidad o proceso de un sistema utilizando un template.	A, B, C, D		A	B	C, D	
Historias de usuario	Con unas pocas oraciones se describe en lenguaje informal o de negocio qué hace el usuario cuando interactúa con la aplicación para cumplir una cierta tarea.	A, B, D	B				A, D



Lenguajes formales	Descripción de funcionalidades mediante lenguajes formales, por ejemplo, fórmulas matemáticas.	A, B, C, D	B	D	A, C		
--------------------	--	------------	---	---	------	--	--

¿Otros? ¿Cuáles?

A: wiki

**Validación de requerimientos**

Técnica	Descripción	Conoce	Nunca	Casi nunca	A veces	Casi siempre	Siempre
Reviews o walk-throughs	Lectura y corrección de la completa documentación o modelado de la definición de requerimientos.	A, B, C, D			C	D	A, B
Auditorías	Revisión de los requerimientos utilizando un checklist predefinido.	A, C	A, C				
Matrices de trazabilidad	Chequear los objetivos del sistema con los requerimientos identificados	A	A				
Prototipos	Aún sin conocer todas las funcionalidades del sistema armar una versión preliminar reducida para que el cliente se haga una idea de cómo será el producto terminado.	A, B, C, D		B	A, C, D		

¿Otros? ¿Cuáles?



6. Técnicas o herramientas de gestión de requerimientos

Escalona describe una serie de técnicas específicas para el tratamiento de requerimientos de aplicaciones web. Identificar si se utiliza alguna ellas.

Actividad / Herramienta	Conoce	Utiliza
Diccionario de datos	B, C, D	C, D
Diagrama de contexto	A, B, C, D	A, C
DFD	C, D	C
Diagrama de clases	A, B, C, D	C, B, D
Diseño de pantallas	A, B, C, D	A, B, C, D
Modelo de datos	A, B, C, D	A, B, C, D
NF. Navegación	A, B, C, D	A, B, C, D
NF. Seguridad	A, B, C, D	A, B, C, D
NF. Mantenibilidad	A, B, C, D	A, B, C, D
NF. Accesibilidad	A, B, C, D	B, C, D
NF. Escalabilidad	A, B, C, D	A, B, C, D
Utiliza algún template	A, B, C, D	C
Utiliza herramienta CASE	A, B, C, D	C
Elicitación y el diseño en paralelo	A, B, C, D	A, C, D



7. Importancia de las características de las aplicaciones web

Kappel habla sobre características relacionadas al producto, al uso y al desarrollo propias de los sistemas web.

Indique según su criterio la importancia de cada característica en el momento de elicitación. (Puntaje de 1 a 5)

Relaciones al producto

De contenido	
Característica multimedia y centrada en el contenido <i>Se tienen en cuenta los generadores de contenido? (estático, dinámico, en vivo)</i> <i>Se tienen en cuenta las diferentes formas de publicación? (texto, animación, video)</i>	A: 5 B: 5 C: 3 D: 3
Demandas de calidad <i>Se tiene en cuenta la calidad del contenido? (frecuencia de actualización, exactitud, consistencia, confiabilidad)</i>	A: 5 B: 3 C: 3 D: 2
De hipertexto	
No linealidad del contenido <i>Se tienen en cuenta las diferentes formas de navegación? (búsquedas, filtros, vínculos)</i>	A: 5 B: 2 C: 3 D: 3
Sobrecarga cognitiva y desorientación <i>Se tiene en cuenta la desorientación que puede causar la no linealidad del contenido? (breadcrumb)</i>	A: 4 B: 3 C: 4 D: 4



De presentación	
Estética <i>Se tienen en cuenta la estética como factor de éxito en un medio donde existe una alta competitividad, modas, tendencias?</i>	A: 5 B: 4 C: 4 D: 5
Ser autoexplicativa <i>Se tienen en cuenta que el usuario debe poder usar la aplicación sin haber leído antes documentación de la misma?</i>	A: 5 B: 4 C: 4 D: 4

**Relacionadas al uso**

Contexto social	
Esponaneidad de los usuarios, carga impredecible y concurrencia <i>Se considera la carga y simultaneidad de usuarios que va a tener la aplicación?</i>	A: 4 B: 3 C: 3 D: 4
Heterogeneidad cultural de los usuarios <i>Se consideran las diferentes culturas y costumbres de los usuarios? (formas de pago, sist medición, idioma)</i>	A: 4 B: 2 C: 3 D: 4
Contexto tecnológico	
Necesidad de calidad de servicio <i>Se tiene en cuenta atributos como disponibilidad, uso de ancho de banda, confiabilidad y estabilidad?</i>	A: 5 B: 2 C: 4 D: 4
Soporte multiplataforma <i>Se tiene en cuenta el entorno operacional del sistema? (diversidad de navegadores, tamaño de pantallas, distintos dispositivos)</i>	A: 4 B: 2 C: 4 D: 4
Desempeño <i>Se tiene en cuenta que el contenido solicitado debe tener un tiempo de respuesta máximo para que el usuario no pierda interés? (5seg, Camus)</i>	A: 5 B: 3 C: 5 D: 3
Vínculo estrecho entre modelo de negocios y arquitectura <i>Se tiene en cuenta el modelo de negocios en el momento de proponer una arquitectura?</i>	A: 4 B: 4 C: 4 D: 4
Arquitecturas abiertas y modulares <i>Se utilizan módulos o componente prearmados?</i>	A: 4 B: 4 C: 5 D: 4



Contexto natural	
Globalidad <i>Se tiene en cuenta la posición geográfica de los usuarios? (huso horario, idioma)</i>	A: 4 B: 3 C: 5 D: 4
Disponibilidad <i>Se tienen en cuenta la disponibilidad que debe tener la aplicación?</i>	A: 5 B: 4 C: 5 D: 4

**Relacionadas al desarrollo**

Equipo de desarrollo	
Multidisciplinario <i>Desarrollo, funcional, marketing, diseño</i>	A: 3 B: 5 C: 3 D: 5
Promedio de edad bajo <i>Menor experiencia en proyectos y mayor interés en nuevas tecnologías</i>	A: 4 B: 2 C: 1 D: 4
Desarrollo comunitario <i>Desarrolladores que no forman parte del equipo, distribuido</i>	A: 3 B: 2 C: 1 D: 4
Infraestructura técnica	
Heterogeneidad <i>Infraestructura y desarrollo</i>	A: 5 B: 4 C: 4 D: 3
Inmediatez e Inmadurez <i>TTM corto hace que el producto se publique inmaduro</i>	A: 3 B: 3 C: 4 D: 4
Seguridad <i>Red pública, almacenamiento en navegador, protección del contenido sensible</i>	A: 4 B: 4 C: 5 D: 4



Proceso	
Flexibilidad <i>Adaptabilidad a cambios en las condiciones del proyecto</i>	A: 4 B: 3 C: 5 D: 4
Paralelismo <i>Desarrollo o pruebas de componentes en paralelo cuando los tiempos son cortos</i>	A: 4 B: 4 C: 5 D: 4
Integración	
Interna <i>Aplicaciones legacy</i>	A: 1 B: 5 C: 5 D: 4
Externa <i>Servicios de terceros</i>	A: 5 B: 4 C: 5 D: 4



Evolución	
Cambio continuo <i>Cambios tecnológicos o funcionales durante el desarrollo de un producto</i>	A: 5 B: 4 C: 5 D: 5
Presión competitiva <i>TTM corto, procesos formales no sirven</i>	A: 4 B: 3 C: 2 D: 4
Paso acelerado <i>Ciclos cortos, sin versiones predefinidas</i>	A: 5 B: 4 C: 2 D: 4
Incertidumbre del cliente <i>Entorno de aplicaciones web hace que el cliente no pueda comunicar efectivamente los requerimientos. Requerimientos guiados por visión más que por necesidad de negocio.</i>	A: 5 B: 3 C: 4 D: 4



8. ¿Cuál es su opinión sobre el proceso de gestión de requerimientos actual?

Entrevistado A

Está conforme con el proceso actual pero siente que todavía hay muchas mejoras posibles. Además considera que la ingeniería de requerimientos debe ir acompañando cambios y el crecimiento de la compañía. La maduración de la compañía debe ser acompañada, de todos modos, la presión por el cambio y la velocidad del proyecto de desarrollo hace que estos cambios deban realizarse de a poco.

Entrevistado B

El proceso de elicitación está bien definido, pero en mi opinión no está bien implementado. Las etapas de aprobación están bien definidas y ejecutadas, pero la gestión de requerimientos depende mucho de quien la realice, y muchas veces se ejecutan de manera informal.

Creo que esto sucede porque el proceso está definido de forma genérica, sin contemplar la variedad de tipos de aplicaciones que se desarrollan en la empresa, y la variedad de clientes, que si bien son internos, suelen ser de distintas geografías.

Entrevistado C

Nuevamente, el entrevistado establece que depende de cuál de todos los proyectos en los que trabajan sea. Con algunos está más conforme que con otros. Pero en general no está del todo conforme con los procesos actuales aunque entiende que responden a una realidad de negocio.

En ciertos proyectos se trabaja con un alto grado de informalidad lo que, para sistemas, representa un problema. El que fuerza a que no se sea tan formal es el negocio casi siempre. El cliente interno quiere obtener lo que necesita lo antes posible aunque esa respuesta rápida a veces atente contra el resultado.

En la opinión del entrevistado algunos procesos no son lo suficientemente duros. Pero reconoce como ventaja que les permite ser más “rápidos”.

Una reunión o una minuta puede ser suficiente para reconocer un requerimiento. Esto



requiere que la gente que captura los requerimientos tenga mucho conocimiento del negocio mientras que una mayor formalidad (“un proceso más fuerte”) permitiría “aislarse un poco más de lo que es funcionalmente el negocio”, realizar una documentación y una validación más formales.

Entrevistado D

Los tiempos de entrega suelen ser cortos, razón por la cual se usan metodologías ágiles. El proceso está bien definido en base a la metodología Scrum, pero es también actualizado frecuentemente.

Los requerimientos se definen en etapas y por varias personas, por lo cual el cliente tiene tiempo de hacer correcciones. Cuando los requerimientos no están claros, o si no se sabe si son factibles, se suelen hacer prototipos.



9. ¿Se le ocurre alguna mejora al proceso actual?

Entrevistado A

Sí. Por ejemplo, dar más visibilidad a clientes y al equipo de desarrollo del estado de los requerimientos. Dar trazabilidad a los requerimientos por algún medio como puede ser alguna herramienta o publicar el backlog. Que haya más accountability.

Entrevistado B

Las mejoras que se me ocurren están relacionadas con lo que mencioné en la respuesta anterior al describir los problemas actuales. Una mejora posible sería tener una definición de procesos específica para distintos tipos de aplicaciones, es decir, para cada área de desarrollo de la compañía.

Entrevistado C

Considera que hay herramientas que pueden ayudar. Manifiesta que en algún momento utilizaron Rational, que es bastante duro y formal. En su opinión Rational ordena mucho la manera de trabajar pero genera una carga de trabajo extra considerable y también quita flexibilidad. Toma Rational como ejemplo pero considera que podría haber puntos intermedios que, sin ser tan rígidos, ayuden a mejorar el trabajo con requerimientos. Incluso sin que tenga que ser utilizando una herramienta específica.

“En nuestro caso, que tenemos muchas aplicaciones y muy variadas, definir un único proceso que se abarcativo a todas las aplicaciones sería buenísimo. Hoy no lo tenemos, pero si lo tuviéramos, y con una cierta flexibilidad, sería buenísimo”. Lo que reconoce, de todos modos, es que un proceso tan abarcativo sería difícil de conseguir. Por ejemplo, al tener requerimientos para SAP, el tipo de soluciones está muy atado al producto, con lo cual los requerimientos se manejan de una manera que tiene a buscar cómo se puede solucionar con la herramienta, mientras que para otros productos, las soluciones pueden ser mucho más abiertas. “Es muy difícil tener un proceso que abarque toda la diversidad”.



Entrevistado D

En algunos proyectos una definición más específica sería útil. Pero cada proyecto y cada cliente es distinto, por lo cual entiendo que el proceso de elicitación debe tener cierta flexibilidad en cuanto al uso de herramientas que se usan. De todas formas, la finalización del proceso debe tener una cantidad mínima de entregables, como lo son actualmente los user stories y los wireframes.

10. Otros comentarios

Entrevistado A

No.

Entrevistado B

No.

Entrevistado C

“Obviamente depende de la empresa, pero muchas veces los requerimientos se terminan definiendo sobre la marcha. Los procesos pueden ayudar, pero hay que tener flexibilidad”.

Entrevistado D

Considero importante tener una metodología base al momento de realizar una elicitación de requerimientos, pero creo que es más importante tener flexibilidad, y saber qué prácticas conviene aplicar y cuáles no, dependiendo de cada caso. Existen factores como la personalidad y conocimientos del cliente, el tipo de proyecto y hasta las tecnologías utilizadas que tienen una fuerte influencia en cómo se elicitán los requerimientos.



Segunda entrevista

1. ¿Cuáles creen que son los criterios más importantes al momento de definir la forma en que se van a elicitar los requerimientos?

Entrevistado A

Estamos trabajando con un framework ágil, con pequeñas definiciones e iteraciones. En cierta forma el requerimiento nunca está acabado a la hora de desarrollarlo, sino que está parcialmente expresado. El criterio de selección para esta forma de trabajo está dado por experiencia previa y por cómo funciona la web y el e-commerce en particular. En los e-commerce el negocio es muy cambiante. Cuando se intenta “congelar el requerimiento” (expresarlo de manera completa y acabada) cambia el contexto y el requerimiento deja de ser válido. Para poder tener más adaptabilidad al contexto buscamos una metodología con la cual tome menos tiempo armar el requerimiento para poder desarrollarlo.

Obviamente, esta forma de trabajo puede tener un “tradeoff” ya que un requerimiento definido de manera inacabada puede resultar en desvíos pero es asumido. El criterio general es buscar un método con el cual el tiempo que lleva definir un requerimiento, y todos los acuerdos a los que hay que llegar para validarlo, no demore lo suficiente como para que cambie el contexto del negocio y el requerimiento deje de ser válido. Además, cuando el requerimiento está más acabado el riesgo crece porque en general comienza a involucrar más cosas y eso lleva a que sea más difícil llevarlo a cabo.

Entrevistado B

Me parece importante que la captura de requerimientos sea en forma presencial, ya que la comunicación será mucho mejor y se podrán captar cosas que de otras formas se pierden.

También es importante que los requerimientos queden bien registrados, pero debe existir un balance entre lo que se acuerda inicialmente y lo que se puede modificar luego.



Entrevistado C

Lo más importante es que nos ayude a dar respuesta al negocio. Tenemos muchas herramientas con las que soportamos a la compañía y lo principal es que podamos responder a sus necesidades. Segundo, pero en la misma línea es la rapidez, para poder dar una respuesta tenemos que convertir rápidamente las necesidades que nos llegan en funcionalidades que queden andando en el sistema. Como tenemos muchas herramientas, sobre plataformas y lenguajes muy distintos, la herramienta que usemos tiene que ser útil para todos estos tipos de proyectos, tiene que ser flexible.

Entrevistado D

El primer factor importante me parece que es tener un contexto, si tenemos una idea general de lo que se quiere hacer, el elicitor puede realizar una investigación previa y de esta forma tener conocimiento del negocio. Esto es importante para saber qué preguntas realizar, ya que como ya mencionamos, la ingeniería de requerimientos muchas veces implica ayudar al cliente a definir qué es lo que quiere construir.

El siguiente factor, que a su vez está relacionado con el anterior, es mantener una buena comunicación, en ambas direcciones, hasta que las partes estén de acuerdo.

Por último, creo que es importante que la definición de los requerimientos no sea exhaustiva, ya que desde la definición hasta su implementación, las necesidades de negocio pueden cambiar, lo que invalidaría los requerimientos. Por esta razón es que utilizamos historias de usuario para la definición de requerimientos.



2. ¿Cómo priorizaría entre siguientes los criterios: flexibilidad, velocidad y facilidad?

Entrevistado A

Coincido con lo que dicen, en cierta forma es lo que expresé en mi respuesta anterior.

La principal razón por la cual elegimos el método que usamos está entre flexibilidad y la velocidad. Primeramente la flexibilidad, darnos herramientas para poder responder al cambio. Luego la velocidad, para poder dar respuesta lo más rápido posible. Por último la facilidad, se busca utilizar herramientas más fáciles de utilizar pero no es una prioridad. La facilidad se piensa más desde la velocidad, que desde la facilidad propiamente dicha.

Entrevistado B

Flexibilidad en el proceso creo que sería lo más importante. El proceso debe ser fácil de ejecutar para evitar que requiera de una persona altamente especializada para realizarlo, y la velocidad creo que sería el menos importante.

Entrevistado C

Entre esas tres, el orden sería, primero flexibilidad, más que nada por la cantidad de proyectos distintos que manejamos. En el segundo lugar ubicaría la velocidad por lo que decía antes de poder responder rápidamente al negocio. Lo último sería que sea fácil. Sería muy bueno que lo que usemos sea fácil, pero es más importante que sea flexible y rápida.

Entrevistado D

Creo que lo más importante es la flexibilidad, tanto en la forma de obtener los requerimientos, como para si es necesario luego modificarlos. La facilidad sería lo siguiente en importancia, al menos lo facilidad vista del lado del cliente, ya que no es deseable complicar al cliente con conceptos técnicos cuando él en muchos caso no lo es. La velocidad es siempre importante, pero en comparación con los otros criterios creo que es el último.



3. ¿Cuáles son los desafíos y problemáticas que encuentran comúnmente al trabajar con aplicaciones web? ¿Estos desafíos se habían identificado como requerimientos? ¿Por qué sí o por qué no?

Entrevistado A

Por dar un ejemplo, en el último período estuvimos trabajando mucho con funcionalidades relacionadas al caché de la aplicación. Los requerimiento detrás estas funcionalidades son disponibilidad y escalabilidad. Especialmente porque tuvimos una cantidad de visitas y ventas mucho mayor porque se iba a lanzar publicidad en la televisión y otros medios.

Estos requerimientos estuvieron de manifiesto hace tiempo, obviamente no de esta manera, sino en lenguaje de negocio. Después fue nuestro trabajo convertir un pedido como “vamos a tener un pico de transacciones necesitamos poder dar respuesta al 100% de las mismas” en requerimientos más concretos y funcionalidades a desarrollar. Hace tiempo que existía este requerimiento pero en algún momento fue perdiendo fuerza, mientras otros requerimientos eran más importantes, ahora frente a la eventualidad de la publicidad mediática volvió a tomar fuerza. El requerimiento de negocio estuvo siempre, pero obtuvo la suficiente importancia y prioridad como para dedicarle tanto trabajo recién ahora. Esto se debe a que el tiempo de respuesta de un e-commerce afecta directamente en el volumen de ventas que recibe el sitio.

Entrevistado B

La gran mayoría de las aplicaciones web que desarrollamos toman datos provenientes de otros sistemas y los exponen a clientes. El acceso a estos datos sería el primer desafío, ya que esto requiere de una integración con otros sistemas. Luego está el problema de la seguridad, ya que los datos que se exponen son sensibles, y normalmente se publican datos que se encuentra en una red interna segura, en una red abierta como Internet.

Estas condiciones son comunes en nuestras aplicaciones web, y por esto suelen ser tenidas en cuenta sin necesidad de que el cliente lo exprese.



Entrevistado C

Al no desarrollar internamente las aplicaciones los problemas surgen fundamentalmente al probar la aplicación. A veces es porque se detecta algún bug en una parte del software o menos comúnmente algún requerimiento que se entendió mal. Cuando es una falla, suele ser más fácil de corregir. Si la falla es porque se entendió mal un requerimiento planteado, suele ser más complicado de resolver. Es por esto que se pone especial atención en la definición de requerimientos.

Otro punto que es problemático es la integración entre aplicaciones. Esto es un requerimiento común en todas nuestras aplicaciones. Lo que suele necesitar es mucho trabajo para coordinar de qué manera se van a comunicar las aplicaciones, qué formato tendrá la información, qué información estará disponible en línea y cuál no.

Entrevistado D

Los desafíos más comunes suelen ser en los cambios o refinamientos de requerimientos, lo que implica que será necesario dedicar un tiempo adicional antes de comenzar con la implementación. Muchas veces, cuando se agrega detalle a un requerimiento, se suelen encontrar problemas técnicos o funcionales que no habían sido contemplados inicialmente.

La compatibilidad de navegadores web es también un desafío, el surgimiento de nuevas tecnologías, como HTML5, permite realizar muchas cosas que no son compatibles con algunos navegadores, por lo que es necesario mantenerse actualizado constantemente de las capacidades de cada uno.

Los desafíos técnicos, como la compatibilidad de navegadores se pueden contemplar en la etapa de requerimientos fácilmente, pero los desafíos relacionados con la interacción con el cliente suelen ser más difíciles y dependen mucho de quien los gestione.