

NEOinsight: Desarrollando un proyecto de movilidad





NEO managing mobility S.L.

**NEOinsight:
Desarrollando un
proyecto de movilidad**

Noviembre de 2018

Copyright 2018



CONTENIDO

1/ Introducción	4
2/ ¿Cómo elegir un proveedor de movilidad?	6
3/ Desarrollando un proyecto de movilidad	13
El análisis funcional: la clave para conseguir lo que queremos	15
El análisis funcional para el uso de una plataforma comercial	17
El análisis funcional de una aplicación a medida para mejora de procesos	20
El análisis funcional de una aplicación para los clientes	21
Integraciones: ¿Cómo vamos a comunicar los sistemas?	24
Las condiciones de aceptación: La clave para decidir el éxito de los proyectos	27
4/ ¿Queremos un producto terminado? Exijamos documentación	29
5/ ¿Tenemos todos los detalles? ¡Adelante!	32
6/ ¿Queremos que no haya sorpresas? Hagamos seguimiento	36
7/ ¿Queremos estar seguros de que todo funciona? Hagamos pruebas	37

1/ INTRODUCCIÓN

En NEO managing mobility estamos tan seguros de que cualquier empresa de servicios con personal en campo necesita aplicar tecnología móvil a sus procesos de trabajo, que queremos aportar, a través de estos NEO insight, cómo hacerlo fácilmente y enseñaros, desde nuestra experiencia, qué pasos se deben seguir para planificar, desarrollar e implementar un proyecto de movilidad en una empresa.

En el documento anterior “Planificando un proyecto de movilidad” exponíamos la importancia de aplicar movilidad a los procesos de negocio de una empresa de servicios para transformarlos y hacerlos más eficientes, y de cómo ofrecer valor añadido al cliente aportándole información transparente e instantánea sobre los servicios, operaciones y procesos que le afectan. En él se explicaron los pasos necesarios para planificar un proyecto de movilidad empresarial.

En este segundo NEO insight nos metemos en harina y os mostramos el siguiente paso: cómo desarrollar el proyecto de movilidad empresarial planificado.



A través de las siguientes líneas facilitamos una guía práctica en la que se recogen los distintos pasos y decisiones que se deben tomar en una empresa para el desarrollo de un sistema de movilidad, minimizando los riesgos y acotando el proyecto a un coste y plazo determinados.

Iremos, paso a paso, por las diferentes fases de desarrollo, por las que, en base a nuestra experiencia, debería pasar cualquier proyecto, incidiendo en las posibilidades u opciones que existen en cada caso y analizando, al detalle, sus ventajas e inconvenientes.

Si ejecutamos los pasos que a continuación indicamos, de forma cuidadosa, estaremos preparados para implantar el sistema de movilidad con ciertas garantías de éxito.

Tras este documento, planeamos publicar otro, el último como guía para analizar lo que hay que tener en cuenta durante la fase de implantación y una vez que el sistema ya está funcionando.



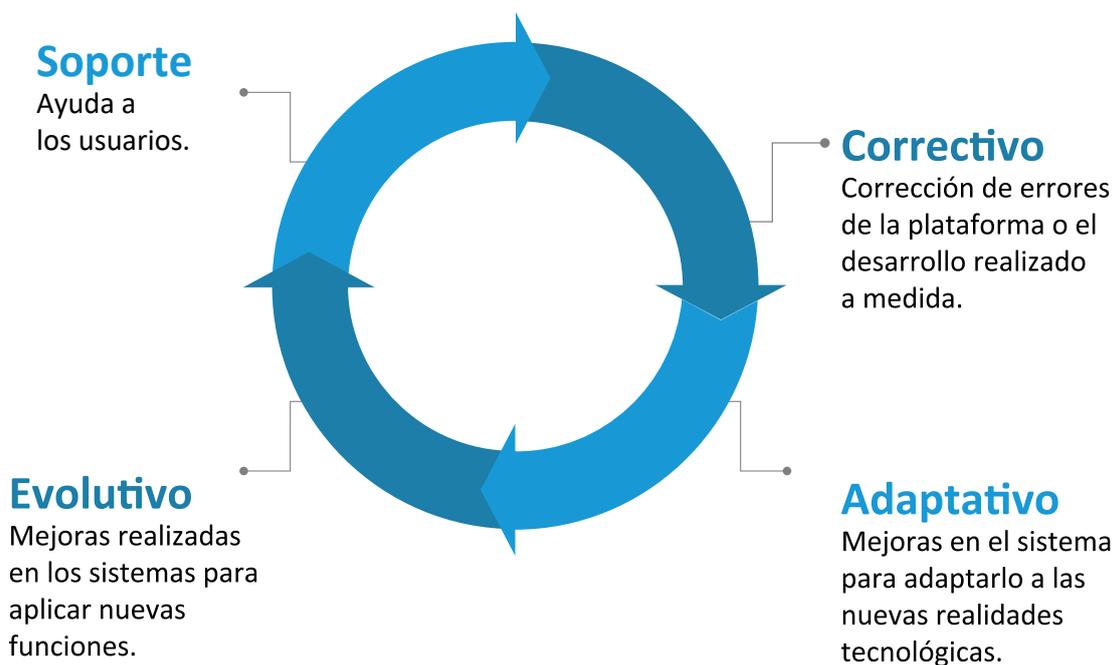
2/ ¿CÓMO ELEGIR UN PROVEEDOR DE MOVILIDAD?

Antes de lanzar un proyecto de movilidad empresarial el primer paso es buscar el proveedor que nos acompañará en el desarrollo del proyecto. Tener el AS-IS (cómo es el proceso de trabajo) y el TO-BE (cómo se desea que sea) realizados correctamente facilitará evaluar la posibilidad de utilizar una plataforma comercial o, en caso de que sea necesario, el desarrollo de un proyecto a medida que cualquier proveedor podrá dar una oferta comprometida (ya que contará con una especificación detallada de lo que se espera del sistema).



Cada oferta deberá detallar los términos de inversión (CAPEX) y mantenimiento (OPEX), procurando no dejar costes ocultos (desplazamientos, pruebas, formación a usuarios, puesta en marcha...).

Dentro del capítulo de costes ocultos es importante evaluar cómo se afrontan los diferentes mantenimientos:



Con estos datos hay que ponerse en marcha y pedir presupuestos. Nuestra búsqueda de proveedores se centrará en diferentes factores dependiendo de si pretendemos utilizar una plataforma de desarrollo o lanzarnos a un desarrollo a medida. En caso de duda, lo recomendable es primero analizar las plataformas disponibles, y solo en caso de que ninguna se acerque a nuestras necesidades, buscar un proveedor de desarrollos a medida.

A la hora de buscar una plataforma de movilidad comercial, tendremos que fijarnos en los siguientes aspectos clave:



1. Versatilidad: La plataforma debe permitir su adaptación a nuestro modelo de negocio, gestionar los datos que manejamos, y ser capaz de modelar nuestros procesos. En el mercado de aplicaciones de movilidad podemos encontrar aplicaciones muy sencillas, que solo permiten diseñar formularios para que se rellenen en el móvil, pero también aplicaciones mucho más potentes que se adaptan al flujo de trabajo de cada empresa. El software workflow con movilidad ofrece la posibilidad de diseñar órdenes de trabajo, despacharlas, planificarlas, y realizar reportes en movilidad con la información siempre accesible en todos los sistemas.

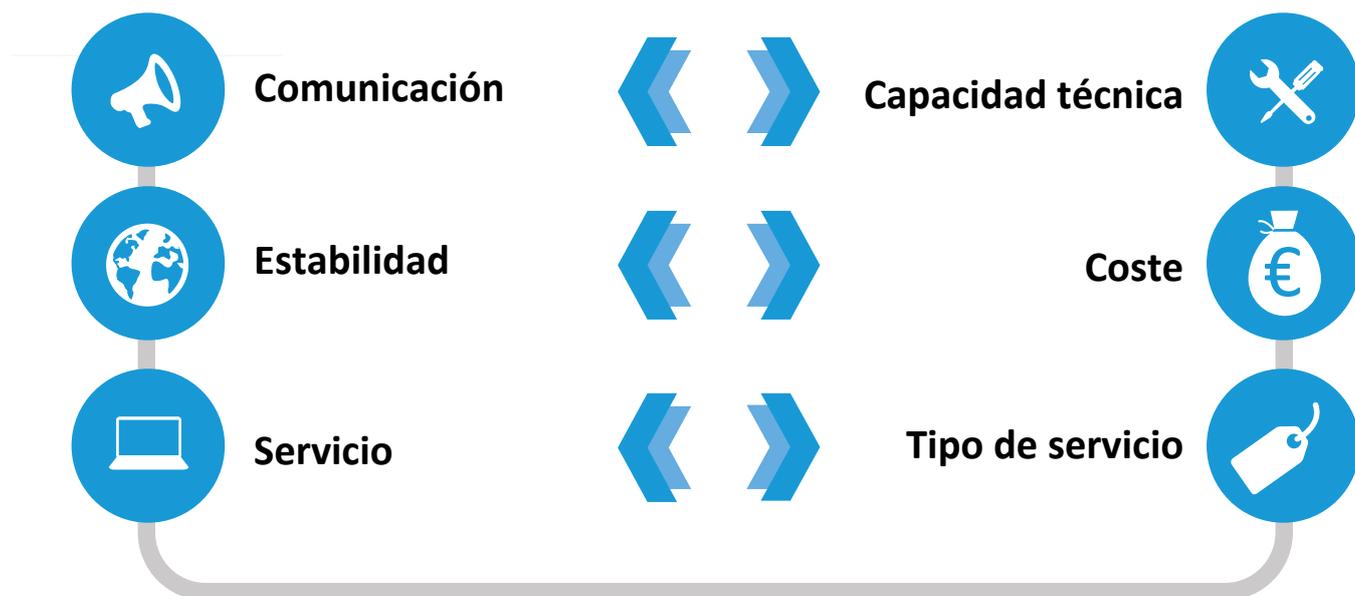
2. Integración: A menudo el proceso que queremos movilizar ya lo estamos gestionando digitalmente en las fases “de oficina”. Si el fabricante de la plataforma está abierto a desarrollar integraciones, puede conseguir que el aumento de eficiencia sea mucho mayor, al evitar el traspaso manual de la información de un sistema a otro.

3. Escalabilidad: Tras implantar una plataforma de movilidad con éxito en uno de nuestros procesos de negocio, es probable que se nos ocurran nuevas mejoras sobre esa base. Estas mejoras pueden consistir en la movilización de nuevos procesos, en la ampliación de los pasos de proceso a movilizar, o en una mayor integración con nuestros sistemas internos de información. Que la plataforma tenga una API abierta (sistema de comunicación con otro software), que no haya limitaciones al número de procesos a movilizar o al número de pasos que pueda tener cada proceso, será fundamental.

4. Costes: Es importante acotar los costes, tanto los iniciales en caso de haber alguna necesidad de integración, como los recurrentes cuando la aplicación esté ya implantada. Los conceptos incluidos en los costes recurrentes pueden ser un factor diferencial clave a la hora de elegir proveedor.

5. Mantenimiento: Al implantar una nueva tecnología, el mantenimiento es una pieza clave del sistema. Que la propia empresa desarrolladora de la plataforma ofrezca atención a los usuarios, ayudando en el proceso de transformación digital, es fundamental para el éxito de la gestión del cambio. Una pequeña sugerencia respecto a tipos de datos, permisos o configuración de los procesos puede ser la diferencia entre un proyecto de éxito y uno que nunca consiga funcionar del todo. Debemos huir de plataformas que tengan el mantenimiento subcontratado a teleoperadores que no vayan a entender la filosofía del proyecto, y valorar especialmente tener acceso al equipo de desarrollo en caso de dudas de alto nivel.

Si optamos por el desarrollo de un proyecto a medida, lo más importante es ajustarnos a la necesidad real. Los criterios a la hora de elegir un proveedor son:



- **Comunicación.** Es importante tener un interlocutor que entienda nuestro lenguaje, al que poder confiar nuestras principales preocupaciones y que nos traslade las dudas del equipo de desarrollo, de modo que nuestras prioridades lleguen adecuadamente al equipo de desarrollo, y aquellos aspectos que pudieran ser susceptibles de interpretación en la documentación inicial se concreten de la forma que nos interesa. Debemos tener en cuenta que la comunicación con el proveedor se alargará durante todo el desarrollo del proyecto, incrementándose en la fase de implantación, y a continuación retomándose según surjan nuevas posibilidades de mejorar el producto.

- **Estabilidad.** Cambiar de interlocutor de forma periódica nos hace correr el riesgo de tener que repetir explicaciones una y otra vez, y una rotación alta en el proveedor nos llevará a acabar en una escalada de excusas respecto a trabajo realizado por una persona que ya no está, suponiendo una mayor frecuencia de incidencias, junto a unos mayores plazos y costes de resolución de las mismas. Con el objetivo de asegurar esta estabilidad, deberemos huir de empresas demasiado jóvenes, que puedan haber hecho una apuesta de negocio demasiado arriesgada, y dejen de estar en el mercado al poco tiempo, dejándonos con un proyecto a medias o sin servicio para el mismo.

- **Servicio.** Es importante que el proveedor entienda cuáles son nuestras necesidades, no solo en el funcionamiento de la herramienta, sino también en las fases de implementación y mantenimiento. Debe ofrecernos un servicio de alto nivel, con comunicación con el equipo de desarrollo cuando las dudas lo requieran. Nuestra relación con el proveedor va a ser muy larga si todo va bien, y un proveedor con servicio técnico subcontratado supondrá un dolor de cabeza cada vez que surjan dudas en el futuro.

- **Capacidad técnica.** Es difícil evaluar la capacidad técnica de un proveedor de desarrollo de software desde fuera, pero ver si su orientación o especialización coincide con nuestras necesidades es siempre prudente. Si necesitamos un proveedor de desarrollo para una aplicación móvil, siempre será mejor un proveedor especializado en movilidad que un proveedor generalista.

- **Coste.** Como en cualquier relación con proveedores, el coste debe ser siempre un factor importante de decisión. Sin embargo, debemos ser cuidadosos a la hora de exigir unos costes o plazos mínimos, ya que corremos el riesgo de contratar un proveedor que los reduzca a cambio de una disminución de la calidad en el proyecto o en el servicio, que lamentaremos posteriormente.

Una duda que podemos tener en este aspecto es la de elegir entre ofertas con un CAPEX (coste de inversión) bajo y un OPEX (coste de mantenimiento) alto frente a ofertas con CAPEX alto y OPEX bajo. Por regla general, el segundo caso demuestra poca implicación del proveedor en el día a día posterior de la puesta en marcha, y puede suponer problemas importantes cuando se encuentran los “problemas de vida real” que no se habían pensado al inicio. Una alternativa ideal es cuando nos ofrecen pagar una parte del desarrollo a lo largo de cierto número de meses de servicio a cambio de un compromiso de permanencia.

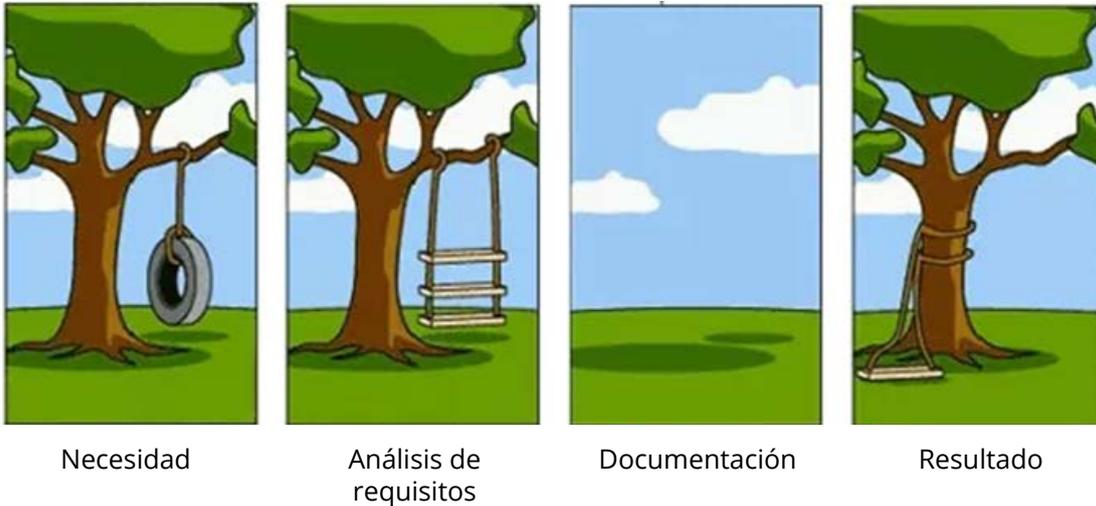
Esto nos permite también una opción muy interesante, que es la de, dentro del proyecto, poner un producto mínimo viable en la calle y luego ir adaptándolo a las necesidades reales del cliente, aunque esto suponga pequeñas desviaciones respecto al planteamiento inicial. La movilidad empresarial tiene una capacidad transformadora enorme, por lo que es preferible salir e ir viendo cómo el negocio se transforma, y adaptando nuestro proyecto en consecuencia, antes que estar un año preparando un proyecto de máxima complejidad que luego no se adapte a las necesidades reales o se haya quedado obsoleto cuando llegue a salir. La rapidez en la salida a producción nos pondrá lo antes posible en contacto con la realidad, permitiéndonos la máxima capacidad de reacción ante las desviaciones encontradas.

- **Tipo de servicio.** Tenemos 2 opciones en este aspecto: servicio SaaS (*Software as a Service*) y servicio *On Premise*. En el servicio SaaS podemos despreocuparnos completamente de la infraestructura, ya que el proveedor pone a nuestra disposición su propia infraestructura desde la nube. En los sistemas *On Premise*, nosotros tendremos que albergar el software en nuestros propios sistemas, y por tanto realizar un mantenimiento que no siempre será sencillo, al tratarse de un sistema desarrollado por terceros. Así, el servicio SaaS es por regla general el más recomendado.



3/ DESARROLLANDO UN PROYECTO DE MOVILIDAD

Con los requisitos identificados y el proveedor elegido, podríamos pensar que es el momento de olvidarnos del proyecto hasta que el proveedor nos informe de que está terminado. Nada más lejos de la realidad. Desentenderse del proyecto durante el desarrollo es garantía de obtener algo completamente distinto de lo que teníamos en mente.



Desafortunadamente, es muy habitual que nos encontremos en una situación similar a las viñetas anteriores. Aunque nuestra necesidad fundamental sea sencilla, a menudo no la transmitimos de una forma completamente precisa, y queremos “aprovechar” para pedir algo más complejo. Normalmente, descuidamos que la documentación vaya mucho más allá de un conjunto de requerimientos de negocio sin concretar los detalles y, finalmente, tras no haber hecho seguimiento ni habernos comunicado con el proveedor durante el proyecto, nos encontramos con una solución que no sirve para nuestro propósito.

En esta fase ya encontraremos una gran diferencia de complejidad dependiendo del tipo de solución que busquemos, y si vamos a desarrollar una aplicación a medida o vamos a utilizar una plataforma de movilidad. En ambos casos, es importante hacer un análisis funcional de lo que queremos, pero en el primer caso es primordial llegar a un acuerdo funcional documentado que llegue a los detalles de lo que necesitamos.

EL ANÁLISIS FUNCIONAL: LA CLAVE PARA CONSEGUIR LO QUE QUEREMOS

Tanto si buscamos implantar una plataforma de movilidad empresarial como si deseamos realizar un proyecto a medida, tenemos que dar un paso de concreción desde nuestro análisis de requisitos. A estas alturas tenemos un TOBE que refleja fielmente nuestros objetivos, pero que se queda normalmente en la generalidad, sin ir al detalle.

Un ejemplo de requisito podría ser “una vez elegida una orden de trabajo, el técnico verá los datos del cliente y del servicio a prestar”. Si dejamos este requisito en manos del proveedor, difícilmente obtendremos la aplicación que queremos, con los datos de cliente que son importantes en nuestro negocio, y los datos de prestación de servicio que nos hacen falta. Un análisis funcional de este requisito podría convertirlo en:

Al presionar sobre una orden de trabajo, al usuario le aparecerá una pantalla con los siguientes datos:

- Nombre y apellidos del cliente (texto, obligatorio).
- Dirección (texto, obligatorio).
- Teléfono de contacto (número, opcional).
- Comentarios del cliente (texto largo, opcional).
- Fecha en que realizar el servicio (fecha, obligatorio).
- Hora en que realizar el servicio (hora, obligatorio).
- Matrícula del vehículo a utilizar (texto, opcional).
- Detalles del servicio a realizar (texto largo, obligatorio).

Este paso es, quizás, el más pesado de llevar a cabo, pero es uno de los más importantes si queremos llevar a buen puerto nuestro proyecto. Tener una descripción detallada de lo que queremos, no tener ambigüedades y estar seguros de que el sistema cubrirá nuestra necesidad es vital, especialmente cuando estamos realizando un proyecto a medida, donde las futuras modificaciones suponen un coste mucho mayor que si se identifica correctamente la necesidad desde el principio.

Lo ideal es desarrollar este punto incluso antes de elegir el proveedor, haciendo que los candidatos lo usen para:

1. Adaptar su oferta a lo que queremos.
2. Proponer cambios y mejoras que aporten valor. La orientación y razonamiento detrás de dichos cambios y mejoras nos permitirá, además, valorar la capacidad del proveedor, especialmente en cuanto a la comprensión de nuestro negocio y nuestras prioridades.

Si el AS-IS y el TO-BE mencionados anteriormente son estudios sobre el negocio, el funcional es ya una documentación sobre el sistema: sobre cómo dará respuestas a las necesidades del TO-BE, cómo serán sus pantallas, qué datos manejará, etc.

Un proyecto saldrá mejor cuanto más completo sea el diseño funcional. De hecho, una buena definición previa permite aumentar en más de un 50% las probabilidades de éxito del proyecto.

La parte central del análisis funcional puede tener diferentes niveles de complejidad según si vamos a utilizar una plataforma comercial, en cuyo caso será un análisis bastante sencillo, o si vamos a desarrollar una aplicación a medida, en cuyo caso el análisis deberá ser mucho más completo, especialmente si nos encontramos ante el desafío de desarrollar una aplicación de cliente, en que tendremos que cuidar hasta el último detalle. Tras el análisis funcional central, será siempre importante en cualquiera de las alternativas dedicar algo de tiempo a pensar en las posibles integraciones a realizar, y en las condiciones de aceptación de nuestro proyecto.

EL ANÁLISIS FUNCIONAL PARA EL USO DE UNA PLATAFORMA COMERCIAL

Este es el caso más sencillo. Hemos elegido una plataforma comercial para movilizar un proceso de negocio, y queremos configurarla de modo que se adapte al mismo. Lo primero que es interesante hacer en este caso es un pequeño gráfico indicando cómo se comporta nuestro proceso de negocio. Pensemos en las personas que están involucradas y las transferencias de información que se producen. Por ejemplo:

- En **atención comercial** recogen los datos del cliente y del pedido. Los almacenan indicando el departamento que debe hacerse cargo.
- En **administración** adjudican el pedido a un coordinador de zona para que se encargue de su gestión. Le pasa los datos recibidos del departamento comercial.
- El **coordinador de zona** busca un empleado disponible para la fecha y hora, llama al cliente en caso de que tenga alguna duda o haya que realizar cambios en la cita. Despacha al técnico la orden de trabajo con su fecha y hora definitiva y los comentarios extra que pueda haber aportado el cliente.
- El **técnico** recibe la orden en su teléfono móvil, través de una notificación. Esta nueva orden de trabajo pasa a su listado de "órdenes de trabajo".

- Cuando el técnico sale hacia el lugar de prestación de servicio, entra en su móvil en el listado de órdenes, elige la orden correspondiente e indica que sale hacia el punto de destino. La información de que ya ha salido queda visible en la plataforma de gestión del coordinador.
- Cuando el técnico llega al lugar de prestación de servicio, indica que ya ha llegado. La información de que ya ha llegado queda visible en la plataforma de gestión.
- Durante la prestación del servicio, el técnico podrá notificar incidencias, quedando estas registradas y visibles en la plataforma de gestión.
- Una vez finalizado el servicio, el técnico rellenará el parte de trabajo, solicitando al cliente su firma de conformidad, y reportándolo para que esté disponible en la plataforma de gestión.

Este proceso se puede representar gráficamente de la siguiente forma:



Este tipo de representación, llamado diagrama de flujo, muestra cómo la información va a fluir entre los distintos participantes del proceso. Nos permite visualizar de forma clara cómo funciona el proceso de negocio y evaluar si nos falta o sobra algún paso (por ejemplo, podríamos añadir un paso final con un supervisor que reporte una evaluación del trabajo realizado). También podríamos pensar en una implantación faseada, y en una primera instancia movilizar solo los dos últimos pasos, que es donde tiene más impacto la movilización, dejando para más adelante la incorporación al sistema del resto de los actores.

Una vez identificados los bloques de información, es importante que paremos a pensar cuál es la información que se va generando en cada paso y propagándose hacia abajo hasta finalizar el proceso de negocio. Para cada uno de esos datos, pensaremos su nombre, su tipo (texto, número, fecha, coordenadas, documento...) y si es o no obligatorio.

Con esos datos, deberíamos ser capaces de configurar nuestra plataforma para que se adapte a nuestro modelo. De hecho, en las plataformas más potentes, denominadas normalmente como de software workflow, podremos dibujar el proceso de forma similar a la gráfica superior, de modo que el propio software se encargue de gestionar el movimiento de la información de un paso al siguiente. En sistemas menos avanzados, tendremos que buscar la forma, normalmente manual, de hacer que la información llegue a su destino.

Llegados a este punto, vemos la gran ventaja (junto con el coste) de haber decidido utilizar una plataforma comercial. No necesitamos hacer una documentación funcional más detallada, no es necesario tener un plan de desarrollo ni hacer seguimiento para evitar desviaciones. Podemos, en definitiva, una vez vistos los apartados de integraciones y condiciones de aceptación en este mismo capítulo, saltarnos los capítulos siguientes hasta llegar al despliegue.

EL ANÁLISIS FUNCIONAL DE UNA APLICACIÓN A MEDIDA PARA MEJORA DE PROCESOS

Si estamos desarrollando una aplicación a medida para mejorar nuestros procesos internos, debemos prestar especial atención al detalle de nuestras necesidades funcionales. El análisis realizado en el apartado anterior sigue siendo una gran herramienta de trabajo inicial, pero además deberemos pensar en las necesidades funcionales, tanto del usuario web como del usuario en movilidad.

Al ser el análisis funcional una tarea relativamente compleja, necesitaremos desarrollarlo de forma coordinada con un analista del proveedor. Los proveedores con más experiencia podrán proporcionarnos un análisis funcional de partida en base a los requerimientos, pero siempre será necesario que nosotros le demos forma y lo completemos con los detalles específicos de nuestro negocio.

Tipo de usuario a tipo de usuario, y requisito a requisito, deberemos evaluar la forma en que cada tipo de usuario podrá cumplir su parte en cada uno de los requisitos. Qué se podrá hacer desde la web, qué se podrá hacer desde el móvil, qué datos se podrán introducir y cuáles serán obligatorios, qué acciones serán necesarias, cómo será la comunicación entre la web y el móvil, que podrá ver cada tipo de usuario...

Así, cada requisito que teníamos se convertirá en un conjunto de funcionalidades detalladas, con flujos de acciones y datos definidos, que articulen el comportamiento que tendrá la aplicación cuando esté desarrollada.

EL ANÁLISIS FUNCIONAL DE UNA APLICACIÓN PARA LOS CLIENTES

En caso de que la aplicación a desarrollar sea para uso de nuestros clientes, tendremos que tener mucho más cuidado con los detalles, definiendo desde una primera fase cómo queremos que sea la imagen que recibirán los usuarios. Tendremos que preparar unos recursos gráficos corporativos, y tener en cuenta que nuestra aplicación va a acabar publicada en las tiendas de Apple y Google. Por ello, aparte de la actividad descrita en el apartado anterior, tendremos que prestar atención especial a lo siguiente:

- **Imagen corporativa:** Nuestra aplicación va a ser nuestro portal en el móvil para nuestros clientes, y por tanto debemos prestar especial atención a que cumpla con nuestro manual de imagen corporativa. El usuario debe ser capaz de entender, de un vistazo, que está tratando con la misma empresa que cuando nos visita en la web o en nuestros establecimientos físicos. Para ello, nos encargaremos de proporcionar nuestros manuales de estilo, logotipo, colores corporativos, tipografía, etc. a nuestro proveedor de software, de modo que sean la base del diseño de la imagen a transmitir en cada una de las pantallas de nuestra aplicación.

- **Icono de la aplicación:** La aplicación que publicamos tendrá entidad propia, de modo que es recomendable que tenga su propia imagen asociada que, si bien cumpla con la estética de nuestra empresa, represente algo distintivo relacionado con la funcionalidad que se espera de nuestra aplicación.

- **Pantalla de carga:** Es fácil olvidarse de la pantalla de carga, dejarla para el último momento y no prestarle excesiva atención. Sin embargo, es la imagen que van a ver nuestros clientes, a pantalla completa, cada vez que presionen sobre nuestra aplicación. La pantalla de carga debe dar una pista de la utilidad de la aplicación, y, sobre todo, transmitir la imagen de nuestra empresa asociada a un concepto de calidad y tecnología.

- **La primera pantalla:** La primera impresión es la que cuenta, y tras la pantalla de carga, nuestros clientes se enfrentarán a unos pocos segundos en los que tomarán una decisión, a menudo definitiva: ¿esta aplicación merece seguir en mi móvil o no le veo la utilidad? Es fundamental que se vea una interfaz fácil de usar, en la que el valor para el cliente se vea en el primer vistazo. Hay que pensar si queremos comenzar con un mapa en el que se ve la ubicación del cliente y los puntos de interés cercanos (si la aplicación tiene como principal objetivo la navegación o la localización geográfica de elementos interesantes para nuestro cliente), o si queremos tener una pantalla resumen de sus datos con un conjunto de acciones a realizar.

- **La primera vez:** Es tentador trabajar siempre en el diseño de una aplicación en base a los datos que tiene un usuario que ya la ha utilizado durante un tiempo. Es bonita, tiene muchos datos, le podemos dar muchas utilidades... pero tenemos que tener en mente que el punto crítico es que nuestro cliente empiece a utilizar la aplicación, y por tanto debemos evaluar cómo se verá cada una de las pantallas cuando el cliente entra por primera vez: que se vea cuál va a ser la utilidad de cada una de ellas y el valor que va a aportar a nuestro cliente, y que llame a la acción dejando claro lo que tiene que hacer para conseguir valor con ella.



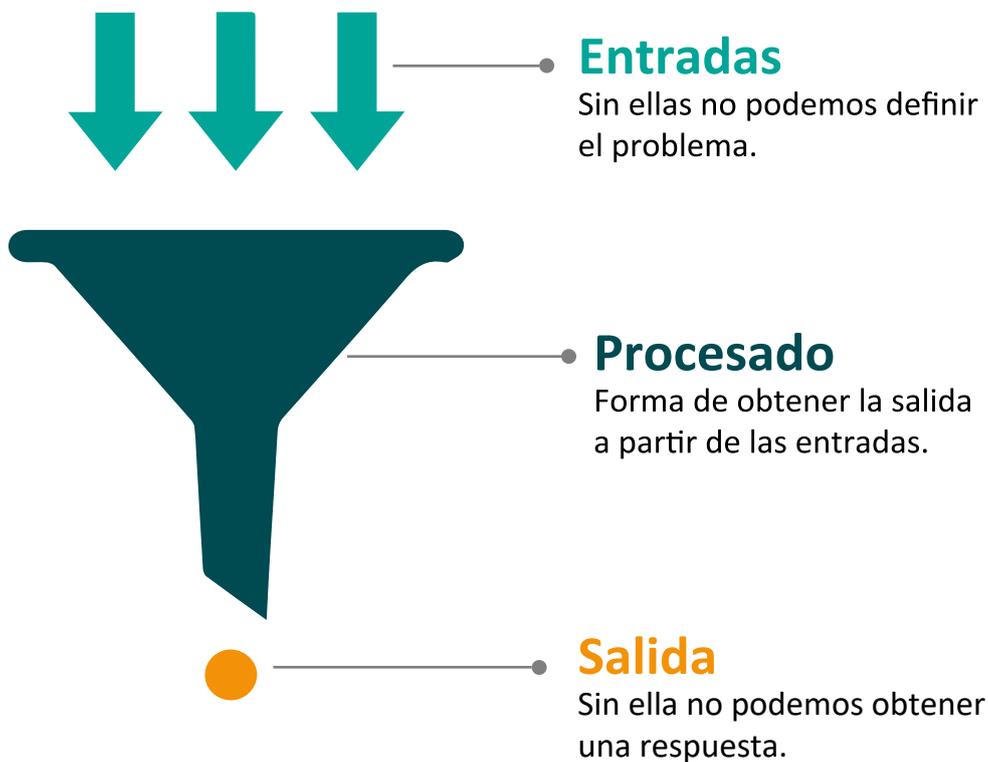
- **La discrepancia entre imágenes, gráficos o texto:** Las imágenes y gráficos en una aplicación móvil ofrecen una connotación de modernidad y de facilidad de uso. Todos sabemos que “una imagen vale más que mil palabras”, pero también tenemos que tener claro que nuestros usuarios no son quienes han diseñado la aplicación, y en muchas ocasiones no van a ser capaces de interpretar para qué sirve un botón representado solo por un icono. Debemos huir tanto de la sobrecarga de texto, que implica una baja usabilidad y ofrecer una imagen de atraso tecnológico, como de su ausencia completa, que acaba siendo confusa para el usuario que aún no conoce la aplicación. El uso de iconos con una o dos palabras descriptoras debajo suele ser un gran compromiso entre ambas posibilidades.

- **Las acciones:** A pesar de todos los esfuerzos que hayamos realizado por simplificar la aplicación, casi siempre habremos acabado con más de tres o cuatro acciones disponibles para el usuario. A la hora de pensar en la aplicación tendremos que afinar por completo: ¿cuáles serán las acciones disponibles para el usuario de forma directa nada más entrar? ¿cuáles llevaremos después de otra acción? ¿cuáles pondremos en un menú? Lo recomendable es que no haya más de cuatro pantallas entre las que navegar con un menú principal, y que en cada pantalla no se puedan hacer más de dos o tres acciones. Todas las acciones accesorias que no se realicen a menudo, como modificación de perfil, cambio de contraseña, etc., deben quedar escondidas en menús menos accesibles para no ensuciar la imagen inicial y dejar claro desde el primer uso a nuestros clientes qué van a poder conseguir gracias a su recién instalada aplicación.



INTEGRACIONES: ¿CÓMO VAMOS A COMUNICAR LOS SISTEMAS?

En este punto ya hemos definido de forma muy clara qué queremos, cuáles son las funcionalidades que va a tener nuestro proyecto, qué datos los van a alimentar, y qué datos vamos a conseguir gracias a él. Nos falta evaluar cómo vamos a introducir los datos en nuestro nuevo sistema, y cómo vamos a explotar los datos que produzca. Esta fase es importante independientemente del tipo de aplicación a realizar.



Así, para cada uno de los datos que alimentarán nuestros procesos, deberemos pensar si ese dato está ya, o lo estamos introduciendo, en alguno de nuestros sistemas informáticos. Si es así, deberemos pensar si a ese sistema le va a seguir aportando valor una vez movilizado el proceso.

Igualmente, para los datos que se gestionan y se producen dentro del proceso, deberíamos evaluar si es un dato que debería acabar en alguno de nuestros sistemas.

En una primera aproximación, pensaremos en el procedimiento de forma manual: las entradas se introducirán manualmente en nuestro nuevo sistema de movilidad, y las salidas de dicho sistema se introducirán manualmente en nuestros otros sistemas. Inmediatamente, es fácil que observemos que se va a desperdiciar mucho tiempo en pasar información manualmente de un sistema informático a otro. Si es así, necesitamos una integración.

Una integración entre sistemas consiste en que, de forma automática, sistemas informáticos distintos se comuniquen información, sin intervención humana y de forma continua. Para ilustrarlo con un ejemplo, utilizaremos el proceso de negocio que queríamos movilizar al comienzo de este capítulo. Los pedidos llegan a través de nuestra atención comercial, y nuestros técnicos reportan sus partes de trabajo. Si tras el proceso de movilización, el personal de atención comercial tiene que introducir los datos de cada pedido en nuestro ERP para que lo tengamos registrado, y en nuestro sistema de movilidad, y luego nuestro personal administrativo, tiene que pasar los datos de los partes

de trabajo del sistema de movilidad a nuestro sistema interno, estaremos perdiendo un potencial muy importante de aumento de eficiencia.

Pensemos por un momento en que aquellos datos que alimentan a nuestro sistema de movilidad y “viven” en nuestros sistemas se comuniquen automáticamente al sistema de movilidad, y los partes de trabajo realizados en el sistema de movilidad lleven sus datos a nuestros sistemas corporativos en el momento en que son reportados por nuestros técnicos. Calculemos las horas de ahorro que podemos conseguir y las ventajas de una comunicación automática sin intervención manual, y por tanto evitando posibilidades de fallo. Si las posibilidades son interesantes, tendremos que estudiar incluir la integración de sistemas en nuestro proyecto.

Si hemos realizado el análisis previo de los datos necesarios para nuestro proceso, la parte inicial de este análisis resultará muy sencilla. Solo tenemos que ver los datos que habíamos detectado que debían alimentar nuestro proceso y evaluar en qué sistemas están dichos datos. Si tenemos control sobre estos sistemas, y podemos solicitar que haya un envío de esos datos a otros sistemas, tenemos la posibilidad de integrar la entrada.

Más complicado es evaluar las posibilidades de integración de las salidas, pero probablemente podamos hacerlo de forma sencilla si estamos trabajando en la movilización de un proceso ya existente. En ese caso, simplemente deberíamos plantearnos que los datos generados en el proceso vayan de forma automática a los sistemas en los que hasta ahora se están introduciendo de forma manual.

Finalmente, debemos evaluar si la implantación de la movilidad nos da alguna ventaja extra que nos permita multiplicar los beneficios. ¿Qué datos nuevos están fluyendo por nuestros sistemas? ¿Qué nueva información está disponible en el sistema de movilidad y nos puede aportar valor?

Estas últimas preguntas, y las obvias ventajas de dividir los problemas en problemas más pequeños para ir paso a paso, hacen que también tengamos que plantearnos una última pregunta: **¿debo realizar las integraciones antes de empezar a trabajar con el sistema de movilidad, o puedo empezar a trabajar con él de forma manual, para en una segunda fase realizar las integraciones correspondientes teniendo ya ejemplos de los datos que el sistema me ofrece, que puedan permitirme optimizar estas integraciones?**

Es importante tener en cuenta que en las integraciones es habitual tener que involucrar a los proveedores de todos los sistemas a integrar, de modo que es más interesante que nunca tener claro el objetivo antes de empezar a desarrollar.



LAS CONDICIONES DE ACEPTACIÓN: LAS CLAVES PARA DECIDIR EL ÉXITO DE LOS PROYECTOS

Un problema habitual al llegar al final de un proyecto es quedarse con una sensación ambivalente: se ha hecho un trabajo muy importante, hay resultados obviamente positivos, pero no tenemos claro si realmente al final tenemos todo lo que queríamos al principio o si nos hemos ido desviando del camino, y finalmente no hemos dado respuesta a las necesidades identificadas. En algunos casos, puede haber hasta discrepancias interpretativas entre la opinión del proveedor y la del cliente. Uno lee los requerimientos y considera que están cubiertos, y el otro no, pudiéndose generar incluso problemas contractuales que conviertan en problemática una relación que debería ser fluida y de confianza.

Para ello, es muy importante desarrollar unos criterios de aceptación, que deben ser objetivos y medibles. En resumen, los criterios de aceptación son unas mediciones que acuerdan cliente y proveedor para valorar que se cumplen todos y cada uno de los requisitos. Por ejemplo, ante el requisito “debe haber pocos campos de formulario para que entren en una pantalla”, podríamos encontrarnos una discrepancia entre el comprador, que lo vaya a utilizar en un terminal pequeño, y el proveedor, que lo valide en un terminal con más resolución. Un criterio de aceptación válido para este requisito sería:

“Se validará con una tablet iPad Mini 4, con configuración de fábrica y en posición vertical que, para los usuarios de tipo técnico, todos los campos de cada formulario entran en la pantalla, viéndose completos ellos y el botón de envío sin necesidad de scroll”.

Si bien en muchos casos los criterios de aceptación de cada requisito pueden parecer obvios para quien lo escribió, es importante hacer un ejercicio de abstracción, entendiendo que el equipo de desarrollo está compuesto por personas ajenas a nuestro negocio, que no tendrán por regla general asimilados conceptos que para nosotros son obvios. Por ello, deberá huirse dentro de lo posible de terminología propia del negocio, y se tratará de concretar adecuadamente cada uno de los criterios. Una técnica habitual para asegurar que siempre ponemos un contexto, un evento y su respuesta o consecuencia, es el formato Given-When-Then (Dado-Cuando-Entonces). El criterio anterior, reescrito para ajustarse al formato sería:

Dado un usuario de tipo técnico con una tablet iPad Mini 4 con configuración de fábrica.

Cuando acceda a cualquier formulario a rellenar con la tablet en posición vertical.

Entonces todos los campos del formulario y el botón de envío se verán en la pantalla sin necesidad de scroll.

4/ ¿QUEREMOS UN PRODUCTO TERMINADO? EXIJAMOS DOCUMENTACIÓN

La documentación es el gran olvidado entre los entregables ya que se echará en falta a posteriori. Un proyecto sin documentar es una bomba de relojería, esperando para estallar en el peor momento: cuando haya que realizar modificaciones, ya sea por algún imprevisto técnico o en el negocio, o porque se quiere expandir la utilidad del proyecto.

Hay 2 tipos de documentación: interna y externa. La primera de ellas es la que solo comparten los partícipes del proyecto y la segunda la que se pone a disposición de usuarios.

Dentro de la documentación interna, cabe destacar:

- **Documentación en el código:** También conocida como “comentarios”. Normalmente no nos llegará, pero es utilizada como una medida clave de la calidad de un código de software. Un código bien comentado es más fácil de modificar y adaptar a las nuevas realidades que se presentan.

- **Documentación de mantenimiento:** Si nuestro proveedor se encarga de la infraestructura y el mantenimiento de la aplicación, esta es una documentación que en principio no necesitaremos, pero si queremos que la aplicación se instale sobre nuestra infraestructura necesitaremos una documentación de inicio, de reacción frente a crash, y de tareas periódicas (como renovación de certificados o verificación de funcionamiento del sistema). También, si vamos a dedicar a nuestro personal a dar soporte a los usuarios, necesitaremos un manual de mantenimiento (a ir expandiendo según lleguen consultas nuevas de los usuarios).

- **Documentación funcional:** el análisis de requisitos y el documento funcional, ya explicados anteriormente, son fundamentales para el acuerdo entre proveedor y cliente en cuanto a las funcionalidades que debe tener el producto.

- **Plan de proyecto:** el análisis de plazos, fases, costes, fundamental para saber cuándo estarán las funcionalidades, cuánto costarán, y poder valorar la afectación de posibles cambios.

Dentro de la documentación externa, tenemos:

- **Manuales de usuario:** es la documentación que necesitan los usuarios de la aplicación para su uso diario. Es conveniente que esta documentación sea concreta y concisa, y que tenga en cuenta el tipo de usuario:

- *Clientes:* Los clientes normalmente no se leerán un manual de usuario. Si se ve que es necesario, probablemente hay que repensar el diseño de la aplicación, mejorando la facilidad de uso, añadiendo ayuda contextual, desarrollando tutoriales...

- *Técnicos de campo:* Los técnicos de campo estarán dispuestos a leerse una documentación breve, pero probablemente solo se la lean una vez, para la primera puesta en marcha. Por ello, su manual de usuario debería ser más un "manual de comienzo rápido", con capturas de pantalla de las pantallas por las que se tienen que mover y una explicación muy breve de las funcionalidades principales.

- *Coordinadores y administradores web:* Son los usuarios en principio más avanzados, y que con más frecuencia podrán recurrir a un manual. El manual debe ser una explicación de cada una de sus actividades y una referencia del significado de cada parámetro de cada menú.

- **Documentación de la publicación:** Es el conjunto de textos, imágenes y capturas de pantalla que se utilizarán para publicar la aplicación en los markets. Es una documentación crítica si nuestra aplicación va dirigida a nuestros clientes, pero puede realizarse de forma menos cuidadosa cuando va dirigida a uso interno. De forma continua Apple y Google van cambiando sus exigencias de datos de cara a la publicación, de modo que lo mejor será que nos asesore el proveedor, y en caso de que la aplicación esté dirigida a nuestros clientes, que encarguemos su contenido a nuestro departamento de marketing.

Es especialmente importante entender que la documentación interna de un proyecto es una **documentación viva**, que no debe quedarse en la primera versión, sino ir adaptándose durante el desarrollo del proyecto, y posteriormente con las modificaciones que se vayan desarrollando.



5/ ¿TENEMOS TODOS LOS DETALLES? ¡ADELANTE!

Una vez que sabemos qué deseamos hacer, que hemos elegido el proveedor y que estamos a punto de ponernos en marcha, hay preparar el plan de proyecto. Es decir, articular cómo vamos a ponerlo en marcha, qué fechas tendrá, qué fases, etc. Y todo ello unido a unos costes y presupuestos asociados.

En caso de que estemos utilizando una plataforma de movilidad comercial, al no requerir desarrollo de código, este capítulo es prescindible, y podemos avanzar directamente al despliegue.

La literatura existente respecto a las distintas fórmulas para llevar a cabo y gestionar un proyecto es muy abundante y diversa y, como clientes, no necesitamos todos los detalles de la mejor forma de llevar un proyecto de desarrollo de software de forma exitosa. Sin embargo, hay algunas decisiones que pueden ser claves en nuestra interacción con el proveedor y en nuestras capacidades para hacer seguimiento y modular la evolución del proyecto:

Las etapas del proyecto

Es tentador entregar unas especificaciones, esperar a la fecha de entrega, y confiar en que entonces tendremos el producto tal y como lo queremos, pero esto no suele funcionar bien. Hay dos razones principales por las que puede ser interesante establecer fases en el desarrollo de un proyecto:

- **En proyectos de larga duración.** Cuanto mayor sea la duración de un proyecto, mayor será la desviación del resultado frente a nuestras expectativas, y mayor el coste de realizar los ajustes correspondientes. Establecer unos puntos de control con entregables intermedios para evaluar que se ajustan a nuestras expectativas y que la dirección que está tomando el proyecto es la adecuada es fundamental para que el rumbo no se pierda. Como norma general, si un proyecto dura más de seis meses, debe ponerse al menos un punto de control intermedio, con sus objetivos parciales a validar. Si es posible porque se abarcan suficientes funcionalidades como para tener un producto útil, debe incluso plantearse la puesta a disposición del producto a los usuarios tras las fases iniciales, de modo que su feedback nos pueda servir de apoyo para reorientar el proyecto y maximizar el éxito final. Es desafortunadamente habitual realizar grandes desarrollos que no se ponen a prueba sobre el terreno hasta el final, cuando ya no hay margen de reacción, y encontrar que los usuarios finales no le encuentran utilidad.

- **En proyectos definidos parcialmente.** Si hay algunas partes del proyecto que tenemos claras, pero hay otras que no tenemos concretadas todavía, y no hay interdependencias que lo hagan inviable, es recomendable poner en una primera fase las especificaciones que tenemos más claras, e ir trabajando en la definición del resto de las especificaciones para lanzarlas en una fase posterior.

La metodología del proyecto

Hay dos metodologías principales en cuanto a su impacto en el cliente: el desarrollo en cascada y el desarrollo ágil.

El **desarrollo en cascada** es el más tradicional. Tras un posible faseado como el indicado recientemente, para cada una de las fases se realiza el análisis de requerimientos y la definición funcional, y se deja al equipo de desarrollo trabajar de forma independiente hasta el final de la fase. Este es el método más recomendable si se pueden definir de forma muy clara los requisitos, si no se pueden ir obteniendo entregables funcionales intermedios o si no tenemos capacidad para hacer un seguimiento continuo del proyecto.

El **desarrollo ágil** consiste en la carencia de un alcance inicial definido. Se hace un esfuerzo inicial mínimo en la definición de los objetivos, y se busca tener de forma casi inmediata un "Producto Mínimo Viable". El Producto Mínimo Viable es el producto que puede estar desarrollado en el mínimo tiempo posible y es capaz de aportar algo de valor a los usuarios. Siguiendo este esquema, en los puntos anteriores solo deberíamos concretar las funcionalidades de este Producto Mínimo Viable, dejando la definición de otras funcionalidades para el futuro, manteniéndolas eso sí, en un listado general, que se denomina *Backlog*.



A partir de un Producto Mínimo Viable, que debe estar disponible para los usuarios, se empieza a funcionar en un sistema cíclico de sprints (períodos de tiempo de dos o tres semanas). En cada sprint:

- Se revisa el backlog para añadir funcionalidades o modificaciones que puedan aportar valor.
- Se ordena el backlog para priorizar aquellos elementos que puedan aportar el valor máximo.
- Se elige un conjunto de elementos que el equipo de desarrollo se vea capaz de producir y poner a disposición de los clientes en el plazo del sprint.
- Se realiza el desarrollo.
- Se valida el desarrollo como paso previo a su puesta en producción.
- Se pone el desarrollo a disposición de los usuarios.
- Se realiza una retrospectiva para evaluar puntos a mejorar en el procedimiento.

En la metodología ágil más extendida (SCRUM), el alcance es fijo durante cada sprint, no pudiéndose añadir o modificar ninguna especificación para permitir la máxima focalización del equipo de desarrollo y evitar tergiversar su compromiso previo al sprint. Sin embargo, en entornos rápidamente cambiantes se pueden utilizar otras metodologías como Scrumban que permiten seguir el modelo anterior y cambiar el alcance cuando corresponda para reaccionar de forma inmediata a los cambios de situación que se produzcan.

La elección entre cascada y ágil debe tener en cuenta los siguientes factores:

- La definición del proyecto. Un proyecto/fase debe estar bien definido antes de empezar si utilizamos el modelo de cascada, ya que no podremos cambiar las especificaciones a mitad del proyecto.
- La duración del proyecto. Si existe un conjunto de funcionalidades que tienen que estar desarrolladas para una fecha definida, el método de cascada nos ofrece esta fiabilidad. Utilizando una metodología ágil, la fecha en que una funcionalidad concreta estará disponible no se puede anticipar. La metodología ágil es ideal, en este aspecto, para proyectos de mejora continua, en los que no es tan importante cumplir unos requisitos en un plazo, como ir mejorando la herramienta para maximizar sus beneficios día a día.
- Nuestra involucración en el proyecto. Un proyecto en cascada nos exige un esfuerzo importante al comienzo, cuando tenemos que definirlo, y posteriormente al final de cada fase, para validar los resultados y lanzar la definición de la siguiente fase. Utilizando metodología ágil, es necesaria la presencia casi continua de una persona que represente los intereses del negocio, para actualizar y repriorizar el backlog de forma continua, y para validar los resultados de cada uno de los sprints. A cambio de este esfuerzo, sabremos que el resultado final estará mucho más alineado con las necesidades de nuestro negocio.

6/ ¿QUEREMOS QUE NO HAYA SORPRESAS? HAGAMOS SEGUIMIENTO

Una vez comienza el desarrollo, si estamos utilizando una metodología ágil, nos hemos comprometido a realizar un seguimiento importante de sprint a sprint, pero eso no quiere decir que en caso de optar por un desarrollo en cascada debamos dejar el equipo de desarrollo desasistido hasta la fase de pruebas de la primera fase.

El desarrollo de un software se parece más a la construcción de una casa que a la de un puente. Cuando hacemos un puente, tenemos muy claras las especificaciones que necesitamos, y durante la construcción es muy difícil que el equipo de ingenieros se encuentre con una incidencia en la que necesite de nuestra ayuda. Sin embargo, al construir una casa, el modelo de persianas que queríamos puede haberse descatalogado, puede ocurrir que el suelo tiene una placa de roca de un material duro que no habíamos previsto y que encarece la construcción de los cimientos, o simplemente durante su construcción puede salir al mercado un nuevo modelo de placa solar que nos permita abaratar costes o tener un mejor rendimiento de la calefacción.

Así, es recomendable programar reuniones periódicas con el proveedor, cada semana o cada quince días, para ir recibiendo actualizaciones sobre el desarrollo y solventando aquellos puntos en los que pudieran surgir dudas o nuevas posibilidades. Esto nos permitirá además estar al tanto de la evolución del proyecto y revisar la planificación de la implantación si es necesario.

7/

¿QUEREMOS ESTAR SEGUROS DE QUE TODO FUNCIONA? HAGAMOS PRUEBAS

Un capítulo especialmente crítico de todo desarrollo, que puede significar el fracaso de un proyecto si no se le presta suficiente atención. Si bien los equipos de desarrollo realizan pruebas unitarias (funcionalidad a funcionalidad) mientras realizan el desarrollo, al final de cada fase del proyecto es necesaria la realización de pruebas integrales, y finalmente, pruebas de aceptación.

Las pruebas integrales deben ser realizadas de forma interna por el equipo de desarrollo, de modo que a nosotros nos llegue un software ya probado, y por nuestra parte, verificaremos que supera los criterios de aceptación que habíamos definido.

También es fundamental exigir pruebas de estrés. Las pruebas de estrés consisten en la simulación del uso del sistema de forma real, para prevenir problemas que puedan surgir cuando los usuarios reales accedan al sistema.

Un problema relativamente habitual asociado a este capítulo es el de los plazos. Es muy tentador, ante incidencias durante el desarrollo, o ante nuevas necesidades de alcance detectadas, reducir la duración del período reservado a pruebas: es el último en el plan, y no tiene normalmente un desglose suficientemente claro, ya que es imposible prever los problemas que se detectarán en las pruebas y el tiempo que se necesitará para resolverlos.

Para hacernos una idea de lo importante de este apartado, deberíamos pensar que la suma de los capítulos de pruebas y documentación debería ser al menos un 50% del tiempo y coste total para garantizar un producto de calidad. Si reducimos ese tiempo de pruebas, nuestros usuarios acabarán siendo los que encuentren los problemas que se encontrarían en esta fase, obteniéndose una imagen de mala calidad del software que puede hacer fracasar el proyecto.

Realizadas las pruebas, ya estamos listos para avanzar, estamos preparados para implantar el sistema de movilidad cuyo proceso desgranaremos en el siguiente NEO insight.

En **NEO managing mobility**, planeamos publicar próximamente sendos documentos para cubrir los puntos a tener en consideración en ambas fases para maximizar las posibilidades de éxito del proyecto.



Oficinas Centrales

Calle Pollensa 2, Edificio Artemisa 17
28290, Las Rozas de Madrid, Madrid.
T. (+34) 91 575 18 06

Operaciones

Calle Pollensa 4, Edificio Atenea 4
28290, Las Rozas de Madrid, Madrid.
T. (+34) 91 636 34 14
(+34) 91 636 34 13

Factoría de Software

Parque Científico de Murcia – Campus Espinardo
Crta. Madrid km 388, CP 30100, Murcia
T. (+34) 96 827 78 34



neo@neo-si.com
www.neo-si.com
www.workandtrack.es
www.workandtrack.mobi