

# Metodologías Agile en el Ayuntamiento de Barcelona

## Programa de digitalización abierta del Comisionado de Tecnología e Innovación Digital del Ayuntamiento de Barcelona

Esta guía ha sido elaborada por un equipo dirigido por Francesca Bria, comisionada de Tecnología e Innovación Digital. Los miembros del equipo son los siguientes: Francesca Bria, Paco Rodríguez, Malcolm Bain, Joan Batlle, Ana Bastide Vila, Xabier Barandiaran Fernández, Marius Boada Pla, Guillem Marpons, Xavier Roca Vilalta, Xavier Bes Segovia, Josep Carles Collazos, Joan Domènech Bas, Oscar Sanz García, Carlos Echevarría Mesegur, LLuis Girona Frisach, Anna Majó Crespo, Tomás Gea Calza, Javier Ruiz, Gemma Galdon, John Michaelides, Esther Bretschneider.



<b>1. INTRODUCCIÓN Y CONTEXTO .....</b>	<b>3</b>
<b>2. PRINCIPIOS GENERALES DE DESARROLLO AGILE .....</b>	<b>4</b>
2.1. Concepto y definiciones .....	4
2.2. Beneficios .....	6
2.3. Implicaciones y retos .....	7
<b>3. AGILE Y ADMINISTRACIÓN PÚBLICA .....</b>	<b>9</b>
3.1. La práctica Agile en la Administración pública.....	9
3.2. Consideraciones importantes .....	10
<b>4. POLÍTICA AGILE DEL AYUNTAMIENTO DE BARCELONA .....</b>	<b>11</b>
4.1. Principios generales y directivos de la política Agile del Ayuntamiento de Barcelona .....	12
<b>5. IMPLANTACIÓN DE LA POLÍTICA AGILE EN PROYECTOS Y SERVICIOS TIC .....</b>	<b>14</b>
5.1. Criterios básicos y estructurales .....	14
5.2. Metodología Scrum@IMI .....	14
5.3. Roles y responsabilidades .....	15
5.4. Actividades .....	16
5.5. Los equipos de trabajo y su gestión .....	17
5.6. Ejecución de proyectos .....	18
<b>ANEXO 1 – MÁS INFORMACIÓN; REFERENCIAS .....</b>	<b>20</b>

# 1 Introducción y contexto

Este documento constituye un documento de apoyo al Código de prácticas tecnológicas del Ayuntamiento de Barcelona, explica de forma práctica el concepto de transformación Agile y expande los principios que guían la nueva dirección del Instituto Municipal de Informática (IMI) en materia de tecnología e innovación: interoperabilidad, agilidad, reutilización, ética y apertura de conocimientos y tecnología. Los principios y las prácticas se derivan del Plan de transformación digital Agile del Ayuntamiento, de septiembre del 2017.

Pretende contribuir y guiar la implementación, por parte del IMI, junto con las gerencias y las unidades de la ciudad, de la intención estratégica del Ayuntamiento de Barcelona de desarrollar proyectos de servicios digitales siguiendo las metodologías Agile. Estas metodologías ágiles, junto con el principio de

soberanía tecnológica y la gestión responsable de datos, componen los elementos principales de la visión a medio plazo para la transformación digital ágil en la ciudad de Barcelona.

El objetivo de este documento, pues, es proporcionar a los responsables de los órganos municipales y del IMI una comprensión básica de los métodos ágiles y demostrar cómo el ciclo de vida ágil es diferente a las aproximaciones tradicionales; explicar cómo se define y cómo se implementará el marco ágil del IMI, integrado en el modelo de proyectos y mantenimiento de servicios digitales existentes; y proporcionar directrices para gestionar proyectos de desarrollo y mantenimiento de servicios basadas en métodos ágiles. De manera general, intenta contribuir a fomentar y gestionar el cambio cultural en el IMI y en las unidades de la ciudad de Barcelona bajo la perspectiva de transformación digital ágil.

# Principios generales de desarrollo Agile

## 2.1. Concepto y definiciones

---

El desarrollo de *software* ágil es una filosofía, y *ágil* un término paraguas para una colección de métodos o aproximaciones que comparten características comunes. Existen diversas metodologías ágiles, de entre las cuales las principales son *scrum* y *extreme programming* (XP).

Generalmente, son un conjunto de metodologías utilizadas en el ámbito del desarrollo y el mantenimiento de *software*, basadas en procesos iterativos y de duración corta (típicamente duran de una a cuatro semanas), que dan como resultado la entrega inicial de un producto parcial, pero operativo, y de versiones consecutivas cada vez de mejor rendimiento. A través de estas iteraciones continuadas, estas metodologías buscan aportar valor desde el inicio del proyecto, así como llevar a cabo una evolución continua del producto. Su objetivo es introducir mejoras y una evolución constante de los productos hasta alcanzar un resultado final excelente que dé respuesta a todos los requerimientos del usuario.

Esta estrategia iterativa permite minimizar el riesgo, ya que cada iteración es como un proyecto en miniatura e incluye todas las fases necesarias: planificación, análisis de requisitos, diseño, codificación, pruebas de usuario y documentación. Así, los problemas en la ejecución, la adaptación a los requerimientos y los riesgos del proyecto afloran con anterioridad y las acciones correctoras son menos costosas y más inmediatas que en un proyecto tradicional de desarrollo (en el que acostumbran a aflorar en su fase final, tras meses de evolución).

Asimismo, las metodologías ágiles se centran en la satisfacción del usuario, ya que requieren que este participe activamente en el proyecto tanto durante su conceptualización como en su desarrollo (mediante la validación de las entregas parciales). De esta manera, se garantiza que el producto final responda a las necesidades del usuario y esté en línea con sus expectativas.

Todo esto se basa en el denominado "Manifiesto por el Desarrollo Ágil de *Software*", de 2001:

### Manifiesto ÁGIL

**Estamos descubriendo formas mejores de desarrollar *software* tanto por nuestra propia experiencia como ayudando a terceros. A través de este trabajo hemos aprendido a valorar:**

**Individuos e interacciones sobre procesos y herramientas.  
Software funcionando sobre documentación extensiva.  
Colaboración con el cliente sobre negociación contractual.  
Respuesta al cambio sobre seguir un plan.**

**Esto es, aunque valoramos los elementos de la derecha, valoramos más los de la izquierda.**

<http://agilemanifesto.org/iso/ca/manifesto.html>

El Manifiesto Agile plantea doce principios que deberían perseguir las metodologías ágiles:

1. Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de *software* con valor.

2. Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.

3. Entregamos *software* funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.

4. Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.

5. Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.

6. El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.

7. El *software* funcionando es la medida principal de progreso.

8. Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.

9. La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.

10. La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.

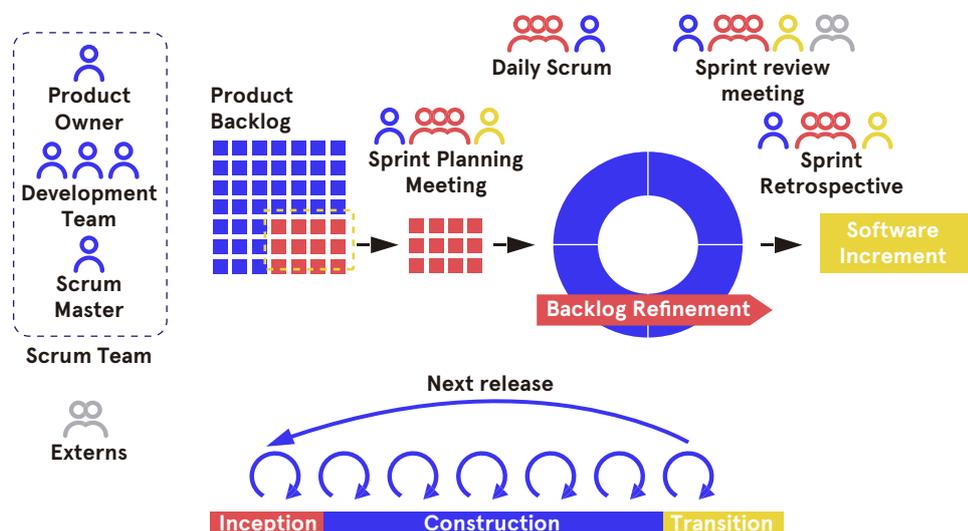
11. Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.

12. A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.

<http://agilemanifesto.org/iso/ca/principles.html>

La clave del modelo Agile es que el proyecto global se divide en una serie de ciclos de desarrollo cortos (conocidos generalmente como "iteraciones" o, en la terminología *scrum*, como "*sprints*") de dos a cuatro semanas aproximadamente cada una. Para cada una de las iteraciones, el equipo de desarrollo lleva a cabo esencialmente las mismas actividades del modelo de cascada (es decir, planificación, diseño, codificación, tests y despliegue), pero bajo prácticas de trabajo ágiles.

El gráfico que se muestra a continuación representa un ejemplo del ciclo de vida de un proyecto, a partir del modelo ágil *scrum*, junto con los actores involucrados en cada una de las fases. A lo largo del documento se irán explicando con detalle los diversos conceptos introducidos en el gráfico.



Las actividades habituales del método ágil scrum son las siguientes:

1. Descubrimiento
2. *Backlog*
3. Iteraciones/*sprints*

La actividad de **descubrimiento** define el producto mínimo viable (PMV) que hay que producir. Se basa en talleres de trabajo y entrevistas que producen una **visión de producto**, que es la definición básica del sistema que se quiere desarrollar.

El **product backlog** ("pila de producto") es una lista priorizada de todos de los elementos individuales estimados para llevar a cabo el proyecto global, es decir, los elementos distinguibles (programables) que componen la visión del producto.

**Para cada sprint** (un ciclo de desarrollo de 2-4 semanas), el equipo de desarrollo prepara un **sprint backlog**. Este documento establece qué elementos de desarrollo del *product backlog* se llevarán a cabo durante el *sprint* en curso

(empezando con los elementos de prioridad más altos). En otras palabras, es una lista "todo" (de cosas que hay que hacer) para el *sprint* en curso.

El equipo de desarrollo diseñará, programará y probará cada uno de los elementos de desarrollo listados en el *backlog del sprint*, durante las 2-4 semanas que este dure. Cabe mencionar también que, típicamente, el equipo se reúne una vez al día para asegurarse de que todas las actividades del *sprint* progresan adecuadamente según se han planificado.

El resultado al final de cada sprint debería ser un *potentially shippable product increment* (un elemento de *software* que puede ser desplegado para el cliente o el usuario), en el que se puede observar un incremento de valor o funcionalidades con respecto al *sprint* anterior.

El conjunto formado por estas tres actividades mencionadas anteriormente define el **ciclo Agile**, que se repite hasta que se obtiene el resultado deseado por el cliente (o al final del presupuesto/tiempo atribuido al proyecto).

## 2.2. Beneficios

---

Entre los beneficios que comporta trabajar con una metodología ágil, podemos destacar los siguientes:

- **Interés en el valor y la satisfacción de cliente**, mediante la entrega rápida y continua de *software* útil. De esta manera, el cliente tiene una visión continua del producto y de su evolución hasta que obtiene el producto final.

- **Interés en las personas usuarias**. Se pone más énfasis en las personas y las interacciones que en el proceso y las herramientas. Clientes, desarrolladores y *testers* interactúan constantemente entre sí para aportar visiones, valoraciones y mejoras del producto.

- **Entregas con más antelación**. El *software* creado se entrega frecuentemente (en semanas más que en meses) a través de los diversos *sprints*, lo que resulta en que haya un

incremento evolutivo en cada uno de ellos con respecto al anterior.

- **Costes y planificación previsible**. El coste es previsible y limitado a la cantidad de trabajo que puede llevar a cabo el equipo, ya que durante las revisiones del *backlog* se calculan los costes de desarrollo de los paquetes de trabajo que lo forman.

- **Compromiso de los stakeholders (grupos de interés)**. El método ágil exige una cooperación próxima y diaria con las personas usuarias, tanto para obtener correctamente los requisitos del proyecto como para verificar el incremento de valor del producto en cada *sprint*.

- **Transparencia**. El progreso en el proyecto es la consecuencia de todo el proceso en sí mismo, y eso hace que los diversos actores involucrados puedan verlo en todo momento.

Además, la comunicación entre estos actores es más directa (“cara a cara”) y constante durante todo el proyecto.

- **Mejora de la calidad.** El método ágil produce una atención continua a la excelencia técnica y al diseño de calidad. Utilizar herramientas específicas y técnicas, como la integración continua o el TDD (*test driven development*), aporta una mejora continua de la calidad del producto.

- **Flexibilidad ante los cambios.** El ciclo iterativo permite una adaptación regular a circunstancias cambiantes, como, por ejemplo, la identificación de posibles cambios o mejoras sobre el producto antes de finalizarlo. Este proceso también permite poder aceptar cambios en los requisitos tardíos.

**“Se aumentó la productividad del desarrollo, se realizó la transparencia de las actividades de desarrollo y se redujo la porción relativa de trabajo administrativo (hasta el 25%). El incremento de la eficacia permite a la Administración desarrollar más servicios digitales con el presupuesto limitado”.**

*Agencia finlandesa responsable de los permisos de conducir, citado en Nuottila et al. (2013), sobre un proyecto Agile en el Dpto. de Transporte.*

## 2.3. Implicaciones y retos

Sin entrar en el detalle de los impactos sobre las organizaciones de las metodologías ágiles y sus diferencias con metodologías tradicionales como la cascada, la implementación de las primeras implica un cambio importante en la gestión de proyectos, la participación del personal del cliente (la Administración pública) y la manera de contratar y entregar los servicios del proveedor y los resultados del proyecto.

Con respecto a la organización de proyectos, se puede resaltar el hecho de que, con esta metodología, los resultados entregados no son necesariamente los finales o los predefinidos al inicio del proyecto, al contrario, el *software* resultante del desarrollo soporta variaciones mediante un proceso iterativo, de acuerdo con las necesidades de las personas usuarias y su respuesta ante las versiones iniciales del producto.

Para el cliente (persona usuaria), *Agile* pone énfasis en la colaboración. A lo largo del ciclo de vida de entrega o iteración, ambas partes (cliente y proveedor) trabajan juntas en cada etapa para alcanzar los objetivos. El esfuer-

zo de la colaboración será mayor durante la fase de diseño, en la que las partes trabajan conjuntamente para determinar los requisitos funcionales y no funcionales (como la experiencia de usuario, UX) de la iteración o el resultado entregable. La contratación ágil requiere, por tanto, una dedicación (de gente y de tiempo) mucho más intensiva por parte del cliente en comparación con la contratación tradicional de cascadas.

Con respecto a la contratación, más allá de los impactos mencionados, la entidad no contrata una solución definitiva y definida (desde el principio), sino que contrata el desarrollo de aspectos de una solución que evolucionan y se incorporan de manera incremental, para alcanzar las necesidades identificadas (*user stories*) y los objetivos generales del cliente. Habitualmente, significa que la manera de contratar servicios tecnológicos implementando una metodología ágil se hace basándose en *time and materials* (contratación basada en el tiempo y los recursos que se utilizarán), y eso puede sufrir variaciones según las características del proyecto.

Además, los criterios de entrega y aceptación de resultados tienen que ser más flexibles, dadas las variaciones de planificación motivadas por cambios de especificaciones funcionales y técnicas que resultan de la retroalimentación de las personas usuarias sobre cada iteración o ciclo de desarrollo.

Por tanto, debemos trabajar en los retos de esta metodología de desarrollo si queremos disfrutar de sus beneficios. El desarrollo continuo de las competencias de los equipos de cliente y proveedor, su implicación

en el proyecto y la gestión de este personal son elementos clave de esta metodología. La definición del alcance y el tamaño de los proyectos y los equipos de desarrollo son otros criterios que hay que tener en cuenta al aplicar esta metodología. En grandes organizaciones con sistemas de información complejos, en las que mayoritariamente se opera con metodologías tradicionales, uno de los retos más importantes es integrar y combinar las nuevas prácticas ágiles en los entornos y los procesos ya existentes.

# Agile y Administración pública

## 3.1. La práctica Agile en la Administración pública

---

La adopción de métodos ágiles ha sido muy lenta en el sector público, teniendo en cuenta que el Manifiesto Agile es del 2001. No obstante, hay varios casos de implementación de metodologías ágiles documentados, como en Finlandia (un proyecto de desarrollo de *software* para gestionar los permisos de conducir) o, más cerca, en Andalucía, con dos proyectos del Ministerio de Cultura y Deportes, *eBOJA* y *TOPOS*. El primero fue un proyecto de desarrollo de sistema para el uso del diario oficial regional, y el segundo, un proyecto de infraestructura para la gestión de los sistemas del ministerio regional. En Estados Unidos, la página web *healthcare.gov* se rediseñó utilizando esta metodología, y el Departamento de Defensa ha reconocido la metodología ágil y sus instrucciones para la adquisición de sistemas (Instrucción 5000.2, de 2015) que la soportan. En el Reino Unido, la Driver and Vehicle Standards Agency, el servicio estatal de conducción, ha implementado con éxito un sistema de gestión de ITV con metodologías ágiles y contratos de corto plazo.

Por nuestra parte, en Cataluña también se ha hecho una prueba piloto para aplicar el *scrum* a un proyecto real del IMI, para identificar las implicaciones que tenía el uso de este "marco Agile" y, sobre todo, para saber cuáles son las carencias actuales, tanto en conocimientos como en infraestructura.

Las conclusiones de estos proyectos indican que la implementación de esta metodología

se hizo con éxito, pero no sin retos, tanto generales como particulares, para las entidades públicas titulares de los proyectos. El enfoque permitió a los equipos elaborar un plan de proyecto inicial que podía guiar el proyecto y también corregirlo de acuerdo con los resultados. También ofrece una visión relevante y útil sobre el estado del proyecto en cada momento, tanto para el equipo como para los responsables de gestionarlo y supervisarlos. Se puede considerar un proceso de aprendizaje, en el que los equipos de personas usuarias y de desarrolladores van mejorando sus habilidades. Adicionalmente, la metodología transforma el esfuerzo de estimación y planificación en un proceso continuo y participativo, que implica a todo el equipo, en lugar de ser una fase inicial del proyecto, propia de otras metodologías tradicionales como la de cascada.

De manera más general, diversas administraciones están en proceso de implementar estas metodologías, como el Reino Unido (<https://www.gov.uk/service-manual/agile-delivery>), Nueva York (<http://playbook.cityofnewyork.us/>) y San Francisco (<http://digitalservices.sfgov.org>), por ejemplo, en la Human Services Agency o en el Departamento de Vivienda y desarrollo comunitario). Estas iniciativas basan la entrega de servicios digitales innovadores, más adaptados a las personas usuarias y a unos costes más reducidos, en metodologías ágiles desde su concepción.

## 3.2. Consideracions importants

---

Tal como hemos descrito en el apartado anterior, "Principios generales de desarrollo Agile", los métodos ágiles tienen diversas consecuencias y plantean retos específicos a su implementación en el desarrollo de servicios digitales para la Administración pública.

Con respecto a la organización de proyectos, se debe asegurar un consenso entre todos los agentes implicados en el proyecto sobre qué método se tiene que utilizar; sobre la definición y la atribución de funciones o roles, en particular entre el departamento o unidad "cliente" (habitualmente, el titular del proyecto o *project owner*), la entidad que gestiona el contrato (en el Ayuntamiento de Barcelona, habitualmente será el IMI) y el proveedor; y sobre cómo llevar a cabo estos roles. Esta selección de metodología y definición de roles es más difícil dependiendo de la complejidad organizativa de la administración destinataria y de la integración de los nuevos servicios con sistemas o plataformas existentes y con proyectos ya en marcha. Las administraciones tienen muchos sistemas y servicios ya implementados, de manera que la coexistencia con los proyectos tradicionales puede ser complicada.

Asimismo, será importante poder disponer de equipos de proyecto estables, tanto desde el lado del cliente/usuario, como del equipo desarrollador. La solvencia técnica, en plazos de conocimiento de los métodos ágiles y la gestión del conocimiento, vuelve a ser un aspecto esencial para garantizar el éxito de los proyectos a corto, medio y largo plazo. Asimismo, en la metodología *scrum*, los equipos ágiles deben coordinarse con los equipos que trabajan sobre proyectos existentes o transversales de la Administración.

Además, será esencial garantizar la dedicación de los equipos "cliente/usuario" (personal de las unidades y entidades de la Administración pública) durante la evolución y el desarrollo del proyecto: no se trata de especificar unas funcionalidades y esperar su entrega. La metodología ágil requiere fomentar una comunicación o una interacción más intensa entre usuario y desarrollador para poder alcanzar los objetivos de las iteraciones. Con Agile, los departamentos usuarios participan activamente durante todo el proceso y los desarrolladores están más presentes en la vida diaria de las personas usuarias para crear la visión de producto y las historias de usuario y recoger retroacción de manera iterativa.

Uno de los aspectos más relevantes para una Administración pública es el proceso y las obligaciones con respecto a la contratación de servicios, que deben cumplir una normativa de contratación pública no necesariamente bien adaptada a estas metodologías. Teniendo en cuenta las características de las metodologías ágiles ya mencionadas, hay que buscar más flexibilidad cuando se contratan servicios tecnológicos ágiles (por ejemplo, dividiendo el proyecto por lotes o actividades, especificando la metodología que hay que utilizar con claridad y las tareas de cada parte, y siendo más flexible en cuanto a las especificaciones definitivas y la entrega y la aceptación de cada fase/actividad o iteración) y en el control de la ejecución del contrato una vez adjudicado.

En conclusión, es importante reconocer la necesidad de trabajar bien el cambio cultural de las administraciones públicas como aparte de la transformación ágil, ya que este cambio suele ser más lento y difícil que en el sector privado.

# 4 Política Agile del Ayuntamiento de Barcelona

Esta sección presenta los principios generales y directivos de la política Agile del Ayuntamiento de Barcelona, elaborados a partir de los estándares de servicios digitales del Ayuntamiento de Barcelona, dentro del mar-

co del Plan de transformación digital para el Ayuntamiento de Barcelona. Los objetivos más relevantes con respecto a las metodologías ágiles son los siguientes:

## • PONER MÁS INTERÉS EN LA PERSONA USUARIA FINAL:

- Dar poder a la persona usuaria final en la toma de decisiones.
- Ser más flexible y entregar más valor.

## • ACORTAR EL TIEMPO DE DESARROLLO:

- Mejorar, a la vez, la calidad técnica.
- Entregar más frecuentemente, de manera fiable, para adelantar el retorno de la inversión.

## • TRANSFORMAR LA CULTURA ORGANIZATIVA:

- Poner énfasis en la colaboración y la transparencia.
- Centrar la orientación más hacia el valor manteniendo el control de los desarrollos.

## • IMPLEMENTAR LA MEJORA CONTINUA:

- Sistematizar la mejora continua basada en ciclos cortos de inspección y adaptación.
- Dotar la organización de agentes activos de la mejora sistémica y de los equipos.

## 4.1. Principios generales y directivos de la política Agile del Ayuntamiento de Barcelona

---

Tal como se definen en el documento “Directrices para la prestación de servicios digitales Agile del Ayuntamiento de Barcelona”, los principios generales en el desarrollo Agile dentro de las entidades del Ayuntamiento son los siguientes:

### A) COLABORACIÓN INTERDISCIPLINARIA

Se configuran equipos multidisciplinares con personas provenientes de diversas disciplinas o entornos de trabajo, en los que todo el mundo está completamente implicado y que incluyen los roles y las especialidades necesarios para desarrollar aplicaciones, servicios y procesos. Solo una o dos personas son directamente responsables, pero el éxito del equipo recae por igual en todos los miembros.

### B) EQUIPOS AUTOORGANIZADOS

Los equipos que se organizan de forma autónoma y flexible pueden crear nuevos enfoques y adaptarse a nuevos retos por sí mismos en su entorno. En consecuencia, los roles y las responsabilidades dentro de los equipos tendrán un cierto nivel de flexibilidad y fluidez de acuerdo con la experiencia, las habilidades y los conocimientos de las personas integrantes del equipo, así como con las necesidades de los proyectos.

### C) COMUNICACIÓN

Se debe establecer un plan de comunicación teniendo en mente la transparencia y las responsabilidades de los agentes implicados. Hay que identificar a las personas usuarias clave de los servicios y los procesos municipales, así como a los actores que participan en la parte técnica. Todos ellos serán los destinatarios de las acciones comunicativas que deberán llevarse a cabo con una frecuencia adecuada a los objetivos y las necesidades de los proyectos. Las conversaciones cara a cara tendrán prioridad con respecto a otros canales de comunicación.

### D) TRANSPARENCIA

Deberían compartirse, de manera abierta y transparente, con los actores clave, el estatus del proyecto, las prioridades, los riesgos, los problemas y los “elementos de bloqueo” eventuales para abordar los retos con rapidez. Y eso se puede hacer con comunicaciones directas pero también utilizando herramientas adaptadas a la metodología ágil, como las tablas Kanban.

### E) MEJORA CONTINUA

Se establecerán prácticas de trabajo colaborativo para facilitar una cultura de aprendizaje y de mejora continua, tanto del servicio prestado como de los integrantes del equipo.

### F) RETROACCIÓN RÁPIDA

El equipo se esforzará por hacer iteraciones tan cortas como sea posible. Las prácticas de equipo se harán de tal manera que faciliten la obtención de la retroacción rápidamente y se incorpore al *backlog* del proyecto.

### G) ACEPTACIÓN DEL CAMBIO

Se entiende que los planes y los proyectos evolucionarán y cambiarán a lo largo de la prestación del servicio. Se establecerán prácticas de análisis y de planificación para reflejar este hecho.

### H) PENSAMIENTO ITERATIVO

Las actividades de análisis, diseño, planificación y desarrollo adoptarán un enfoque iterativo en ciclos cortos para permitir la retroacción y cambiar las prioridades según las necesidades.

### I) DESPLIEGUE RÁPIDO, DESPLIEGUE FRECUENTE

Hay que poner énfasis en dar valor a las personas usuarias finales tan rápidamente como sea posible, reducir el tiempo de puesta en producción de las aplicaciones y recibir re-

troacción de las personas usuarias lo más rápido posible. Se dará prioridad al producto mínimo viable identificado que responda a las necesidades prioritarias de las personas usuarias y las entregas siguientes se realizarán de acuerdo con este enfoque.

## **J) INTERÉS EN OFRECER VALOR A LA PERSONA USUARIA**

Se debe dar prioridad a las necesidades de la persona usuaria. Las necesidades de la persona usuaria van por delante de cualquier otra necesidad, incluso de las de los actores clave. Cuando haya un conflicto de necesidades entre los grupos, hay que asegurarse de que hay unas prioridades claras. Se promoverá la diversidad en los equipos para favorecer diseños inclusivos.

## **K) CALIDAD**

La calidad no hace referencia solo a la calidad técnica, sino también a la calidad del servicio. La calidad está presente en cada paso del proceso y todas las personas del equipo tienen la responsabilidad de garantizar que el servicio es de un nivel alto.

## **L) MEDIDA DEL PROGRESO**

La entrega de *software* ejecutable es la principal medida de progreso. También hay que aplicar otras medidas que reflejan el valor aportado a la persona usuaria y al negocio para garantizar que satisface las necesidades reales y da un valor real.

Estos principios y directrices se aplicarán en proyectos y servicios TIC a través de las prácticas y las medidas que se presentan en la sección siguiente.

# Implantación de la política Agile en proyectos y servicios TIC

Para la aplicación de las directrices enunciadas en el apartado anterior, el IMI ha desarrollado una metodología Agile basada en *scrum* y

adaptada a sus funciones como responsable de las TIC del Ayuntamiento de Barcelona.

## 5.1 Criterios básicos y estructurales

---

El IMI llevará a cabo los proyectos de manera alineada con la estructura organizativa del Ayuntamiento de Barcelona.

se haga en modo ágil o en modo tradicional, pero no combinando ambos modelos.

Aunque es viable combinar desarrollos ágiles y proyectos con metodologías tradicionales (enfoque "bimodal"), es más difícil para gestionar las dependencias de los desarrollos con diferentes ciclos de vida y con otros roles externos. Por este motivo, se recomienda que la gestión de los proyectos de las unidades del Ayuntamiento (áreas, sectores y distritos)

Además, dentro de una unidad municipal en la que se llevan a cabo todos los desarrollos en modo ágil, puede ser difícil absorber los picos de demandas sin sobredimensionar la capacidad de los equipos ágiles o sin alargar en exceso los tiempos de respuesta. En este caso, puede ser conveniente contratar proyectos tradicionales en paralelo.

## 5.2 Metodología Scrum@IMI

---

Los proyectos determinados como ágiles seguirán la metodología de desarrollo ágil de aplicaciones del IMI, denominada Scrum@IMI. Está basada en el marco de trabajo *scrum* y tiene en cuenta prácticas de ingeniería provenientes de otros modelos, como DevOps.

- Planificación del desarrollo (*releases, sprints, paquetes de trabajo, defectos, etcétera*).
- Repositorios de documentación, código y binarios.
- Gestión de requisitos y pruebas.
- Automatización de las pruebas unitarias y funcionales.
- Integración y despliegue continuos.
- Control de la calidad del código.

Esta metodología se soportará bajo el uso de una plataforma ALM (*application lifecycle management*) con herramientas que incluirán, entre otros, los siguientes aspectos:

Su uso será obligatorio por parte del adjudicatario sin que le suponga un coste adicional en licencias.

Las características principales previstas para esta metodología se comentan según su ciclo de vida en los apartados siguientes.

Toda la documentación que se genere internamente en el desarrollo deberá gestionarse con las herramientas que se determinen al inicio del proyecto, preferentemente en formato wiki.

Los acontecimientos o *events* (actividades e hitos) habituales de la metodología Scrum@IMI incluyen los siguientes pasos:

Reunión/acontecimiento	Objetivos
<b>Sprint</b>	Ciclo de desarrollo, con una duración de dos a cuatro semanas, en el que se entrega una parte del sistema (incluidos todos los subproductos, como documentación y pruebas). El cliente puede optar por publicar o no el incremento entregado por el proveedor.
<b>Planificación de sprint</b>	Determinar, entre el <i>product owner</i> y el equipo de desarrollo, la parte del producto que se hará durante el <i>sprint</i> .
<b>Daily Meeting</b>	El equipo de desarrollo hace el seguimiento del <i>sprint</i> y las correcciones adecuadas. En caso de desviación, se avisa al <i>product owner</i> y al <i>scrum master</i> .
<b>Revisión de sprint</b>	El equipo <i>scrum</i> , con los actores externos invitados, examina el resultado del <i>sprint</i> y revisa el plan para los <i>sprints</i> siguientes.
<b>Retrospectiva de sprint</b>	El equipo <i>scrum</i> examina su funcionamiento propio y planifica mejoras concretas.

## 5.3 Roles y responsabilidades

La implementación de los roles de *scrum* en el IMI se hace respetando al máximo los objetivos y las responsabilidades estándar de los roles, a la vez que se adaptan al contexto del IMI, que provee servicios complejos de gestión de TIC para el Ayuntamiento de Barcelona.

Los roles principales son los siguientes:

- **Product Owner (Ayuntamiento)**
- **Proxy Product Owner (IMI)**
- **Scrum Master (IMI o proveedor)**
- **Equipo de desarrollo (proveedor)**

### **PRODUCT OWNER (AYUNTAMIENTO)**

El *product owner* (PO) es el rol que tiene como objetivo maximizar el valor entregado a los usuarios. Las mejores implementaciones del rol son aquellas en las que el PO está cerca de donde se toman las decisiones "de negocio", razón por la cual con este rol se apodera a alguna persona clave de los sectores y las gerencias del Ayuntamiento. Este apoderamiento le permitirá poder conocer con transparencia y frecuencia el estado de los desarrollos, para poder controlarlo y tomar decisiones informadas.

El PO del Ayuntamiento típicamente tiene una alta carga de trabajo y muchos interlocutores con quienes trabajar. Esto hace que pueda tener poca dedicación a las actividades habituales del PO, como gestionar y desglosar

el *backlog* del producto y dedicar tiempo al equipo de desarrollo. Por estos motivos, será frecuente que los PO tengan un rol asistente en el IMI, llamado *proxy product owner*. Esto, en ningún caso, debe restarle transparencia en las actividades del equipo ni en la capacidad de decisión.

### **PROXY PRODUCT OWNER (IMI)**

El *proxy product owner* (PPO) no es un rol estándar de *scrum*, pero sí un patrón en situaciones en las que es difícil encontrar a un PO que tenga a la vez visión y capacidad de decisión desde el punto de vista de negocio, o bien también por razones de proximidad al equipo de desarrollo y de disponibilidad para estructurar y priorizar el *backlog* detalladamente. En el contexto del Ayuntamiento y del IMI, será frecuente que el IMI aporte este rol.

### **EQUIPO DE DESARROLLO (PROVEEDOR)**

El equipo de desarrollo, formado por un grupo de tres a nueve miembros (preferiblemente con

perfiles polivalentes), lleva a cabo todas las actividades necesarias referentes al desarrollo del proyecto, con el objetivo principal de la mejora continua. Para hacerlo, el equipo interactúa con todos los roles del IMI necesarios, de los cuales el PO y el PPO serán los interlocutores principales con respecto al trabajo que se debe hacer, y el *scrum master* les servirá de guía metodológica y también de orientación organizativa dentro del IMI.

### **SCRUM MASTER (IMI/PROVEEDOR)**

El *scrum master* (SM) tiene la misión de enseñar *scrum* y utilizar el marco de trabajo para mejorar el valor y la efectividad del desarrollo del equipo. En el contexto del IMI, esto lo consigue haciendo de guía de la metodología, llevando a cabo el seguimiento y los controles del rendimiento del equipo o bien, si hace falta, haciendo un entrenamiento individual a algún miembro del equipo. Adicionalmente, la figura del SM ayudará a hacer que el equipo se integre en torno al IMI y el Ayuntamiento, especialmente con los departamentos transversales.

## **5.4 Actividades**

---

A continuación, identificamos los objetivos y el contenido de las actividades principales de la metodología Agile del IMI. En caso de se deba profundizar más detalladamente en alguna de estas actividades, se puede consultar la documentación proporcionada por el espacio Agile.

### **1. ACTIVIDAD: PLANIFICACIÓN DE ENTREGAS**

La actividad de planificación de entregas se hace previamente a la construcción del producto, pero se puede planificar de nuevo durante esta actividad cuando se considere necesario. Su objetivo es actualizar y controlar la planificación de los *sprints* y las entregas, de manera consensuada entre todo el equipo

*scrum*, con el liderazgo del *product owner* y del *proxy product owner*. El resultado de esta actividad es el plan de desarrollo, basado en la propuesta del adjudicatario a su oferta, y tendrá que ser conforme a los requisitos especificados en el pliego.

### **2. ACTIVIDAD: REFINAMIENTO DEL BACKLOG**

La actividad de refinamiento del *backlog* se hace durante el desarrollo del producto y la lideran los roles de *product owner* y *proxy product owner*. Su objetivo es analizar funcional y técnicamente los paquetes de trabajo del *backlog* del producto.

Durante esta actividad se espera una alta interacción con las personas usuarias y otros roles del IMI. Es recomendable elaborar maquetas estáticas o prototipos dinámicos que incluyan las funcionalidades más importantes del sistema para que el usuario pueda validarlas.

### 3. ACTIVIDAD: SPRINT

El objetivo de esta actividad es la construcción del incremento del sistema, representado por el *sprint backlog*, correspondiente a la prioridad de negocio y técnica que consensúen el *product owner* y el *proxy product owner* con el equipo de desarrollo. Al final del *sprint*, el PO y el PPO son los responsables de hacer la validación y la aceptación formal de los paquetes de trabajo que formen el incremento, siempre con la asistencia del *scrum manager*.

El incumplimiento por parte del adjudicatario al entregar los incrementos realizados al final del *sprint* podrá ser objeto de aplicación de sanciones por parte del IMI, tal como se detalla en la cláusula "Penalizaciones del pliego administrativo".

### 4. ACTIVIDAD: TRANSICIÓN

El objetivo de esta actividad es la entrega y la puesta en producción de los paquetes de trabajo que determine el *product owner* de manera alineada con las necesidades del usua-

rio y otros actores del IMI, como los grupos de operaciones y de servicio de asistencia al usuario. Esta entrega puede ser puntual y bajo demanda durante el *sprint* (por ejemplo, para solucionar una incidencia urgente), o se puede hacer al final del *sprint*.

La aceptación de la actividad de transición está condicionada a la validación y aprobación, por parte del *proxy product owner*, del IMI. En caso de cambios de versiones significativos, la aceptación formal del proyecto se hará en una reunión del comité de dirección del proyecto. El hecho de no alcanzar los objetivos de esta actividad supondrá parar el proyecto.

### 5. ACTIVIDAD: OPERACIÓN

El objetivo de esta actividad, que será de interés para el adjudicatario en el caso de que el alcance de los servicios contratados incluya el apoyo al usuario, es dar apoyo de nivel 2 y 3 (conceptos de ITIL) al grupo de Servicio de Atención al Usuario (SAU) y a la Dirección de Operaciones.

El *proxy product owner* actuará como "responsable del servicio" cuando deba priorizar las incidencias identificadas y escaladas por los grupos de SAU y Operaciones y darles seguimiento. La sección de Apoyo a Usuarios describe los protocolos y los acuerdos de nivel de servicio (ANS) que determinan la prestación de este servicio.

## 5.5 Los equipos de trabajo y su gestión

---

El modelo *scrum* se puede implementar con equipos contratados de dos maneras: equipos para el desarrollo de nuevos proyectos y equipos de mantenimiento de aplicaciones (AM). A continuación, se pueden ver las diferencias que tienen en el contexto Agile.

Los equipos multidisciplinares deberían tener competencias para poder llevar a cabo las siguientes acciones:

- Diseñar y construir un servicio digital.
- Operar y mantener un servicio digital.

Las competencias que se necesitan cambiarán durante el ciclo de vida del servicio y podrán recibir el apoyo de roles adicionales.

Todo el equipo y, en especial, los diseñadores, los investigadores de usuario y los desarro-

lladores deben trabajar juntos para diseñar, construir y probar y entregar el producto.

## 1. EQUIPOS PARA PROYECTOS NUEVOS

Los equipos para proyectos nuevos se deben contratar con un alcance del servicio definido antes de iniciar el desarrollo. La definición del contrato requiere una conceptualización previa del sistema que se quiere construir, que se desarrollará conjuntamente entre el PO del Ayuntamiento y el PPO del IMI. Esta conceptualización debe incluir la creación de los siguientes elementos:

- Un *backlog* inicial que defina el alcance del desarrollo en forma de ítems o paquetes de trabajo.
- Un plan de entregas que identifique las entregas que se harán y qué *sprints* forman parte de las mismas.

## 2. EQUIPO DE MANTENIMIENTO DE APLICACIONES (AM)

Los equipos de mantenimiento (AM) estarán contratados a priori e irán recibiendo peticiones de diversos tipos (correctivas, evolutivas y perfectivas), que se acumularán en el *backlog*. Las peticiones urgentes se atenderán cuanto antes mejor, siguiendo los objetivos del ANS. El resto de peticiones se planificarán para que se atiendan en *sprints*, probablemente

te de una duración más corta (por ejemplo, una semana) que los equipos de proyecto, ya que los paquetes habitualmente serán más pequeños e independientes.

## 3. COEXISTENCIA ENTRE LOS EQUIPOS DE PROYECTO Y LOS AM

La coexistencia entre los equipos *scrum* para desarrollar nuevos productos y los equipos *scrum* que los mantienen tendrá diversas características dependiendo del grado de libertad de los contratos con los proveedores y de la opción que consideren más oportuna el *proxy product owner* y el *scrum master*. Las opciones, en orden de más a menos ideal, son las siguientes:

- El equipo de producto tiene integrados los miembros del AM, de manera estable, para trabajar conjuntamente en las mismas aplicaciones.
- El equipo AM dispone de una dedicación temporal asignada para trabajar en el equipo de producto y preparar la transferencia de la aplicación una vez entregada al IMI.
- El equipo AM no tiene una dedicación asignada para trabajar durante el desarrollo de la aplicación, pero tiene los incrementos frecuentes y los "hechos" para ir preparando la transferencia, y también podría participar puntualmente en las revisiones de los *sprints*.

## 5.6 Ejecución de proyectos

---

Los proyectos Agile gestionados por el IMI se deben ejecutar conforme a los principios siguientes, reflejados en el contrato correspondiente con los proveedores:

**1)** Se seguirán las recomendaciones y las directrices metodológicas ágiles del IMI, como PPO, presentadas en el pliego y dentro del espacio Agile del IMI, así como las que sean contextuales y que transmita el *scrum master*,

que observará un principio general de respeto a la autogestión del equipo.

**2)** Habrá que tener disponible, al final de cada *sprint*, un entregable o incremento de producto pactado (*software* de incremento) que cumpla con los mínimos de calidad definidos en el espacio Agile. En caso de que sea imposible entregar todos los contenidos planificados, prevalecerá la calidad sobre la cantidad.

3) Habrá que ser transparente con el *scrum master* (y eventualmente con el PPO) ante los posibles problemas internos del equipo, así como ante los impedimentos externos, para buscar conjuntamente las mejores soluciones a los problemas puntuales y estructurales del desarrollo.

4) El proveedor utilizará las herramientas corporativas, así como las normas metodológicas mínimas que favorezcan la interoperabilidad entre equipos, como el idioma, la estructura y el nivel de detalle de la documentación del código y otros apoyos documentales.

5) El equipo responsable de los niveles 2 y 3 de la gestión de incidencias del servicio deberá tener conocimientos mínimos de sistemas y trabajar con el IMI de la manera más integrada posible. En un futuro se prevé que, en función de las capacidades de automatización del despliegue de la plataforma usada, el equipo podrá desplegar el sistema de manera autónoma en los entornos de preproducción y producción.

# Anexo 1

## Más información.

## Referencias

### Documentación del Ayuntamiento de Barcelona

Ayuntamiento de Barcelona: "Directrices para la prestación de servicios digitales Agile", 2017.

### Referencias:

Manifiesto Agile: <http://agilemanifesto.org/iso/ca/manifesto.html>

Principios Agile: <http://agilemanifesto.org/iso/ca/principles.html>

### Scrum

#### • **The Scrum Guide, Schwaber y Sutherland**

Este documento de diecisiete páginas es la definición oficial de *scrum*. Se recomienda encarecidamente leerlo para entender la filosofía de este marco de trabajo, así como sus componentes y reglas.

#### • **Scrum Pocket Guide, Verheyen**

Este librito, de aproximadamente 100 páginas, es una ampliación comentada de "The Scrum Guide". Se recomienda leerlo si se quiere tener un buen conocimiento teórico del marco de trabajo.

#### • **Large-Scale Scrum, Larman & Bodde**

Este libro describe el modelo large-scale *scrum* (LeSS) para aplicar *scrum* a equipos múltiples, por lo que contiene contenidos avanzados. No obstante, también revisa de manera reflexiva y detallada los conceptos básicos de *scrum*, además de introducir muchos otros conceptos más avanzados.

### Definir productos y backlogs

#### • **User Stories Applied, Cohn**

Este libro es una guía práctica para entender qué son las historias de usuario y cómo hacer un buen uso de las mismas.

#### • **User Story Mapping, Patton**

Este libro profundiza en la utilidad de la técnica del mapeo de historias para entender mejor los problemas del usuario y aprender a diseñar planes de desarrollo rápidos y que aprovechen las entregas frecuentes para optimizar el valor que se ofrece a las personas usuarias.

### Planificació àgil

#### • **Agile Estimating and Planning, Cohn**

Este libro es una guía práctica sobre el proceso de estimar el esfuerzo de los backlogs de producto y sobre cómo utilizar estas estimaciones para planificar versiones y sprints y hacer el seguimiento de los planes.

- **Scrum & XP from the trenches, Kniberg**

Este libro se puede descargar libremente de la web InfoQ. Es una guía completa y pragmática sobre cómo hacer un desarrollo utilizando *Programmingscrum* y *extreme programming*.

- **Agile Project Management with Scrum, Schwaber**

Este libro recoge la experiencia práctica del cocreador de *scrum* en su aplicación en diversas empresas y explican lo que fue bien y lo que no funcionó.

### Product Owner

- **AGILE Product Management with Scrum**

Este libro hace una buena introducción al rol de *product owner*, a sus responsabilidades y a cómo llevar a cabo las actividades habituales durante el ciclo de vida de un producto.

- **Scrum Product Ownership, Galen**

Este libro explica cómo desarrollar el rol de *product owner*, tanto en organizaciones pequeñas como en organizaciones grandes, e incluye especialmente los aspectos de organización con todos los actores involucrados.

### Scrum Master

- **Scrum Mastery, Watts**

Este libro describe, de una manera ágil y amena, el enfoque correcto del rol del *scrum master*. Es una introducción muy recomendable.

- **Coaching Agile Teams, Adkins**

Este libro es una referencia avanzada para cualquier coach y *scrum master* profesional. Es muy detallado y contiene conceptos avanzados.

- **Succeeding with AGILE, Cohn**

Este libro explica patrones para extender la agilidad en las organizaciones. No hay ninguna metodología unificada para introducir la agilidad en las empresas; estos consejos pueden ser útiles para aplicar al contexto único de cada organización.

## Otra documentación consultada

Gobierno del Reino Unido. "Agile Delivery", <https://www.gov.uk/service-manual/agile-delivery>

Nuottila J., Aaltonen K., Kujala J. "Challenges of adopting agile methods in a public Organization", International Journal of Information Systems and Project Management, vol. 4, n.º 3 (2016), págs. 65-85, DOI: 10.12821/ijispm040304.

Torrecilla-Salinas C. J., Sedeño J., Escalona M. J., Mejías M. "Agile in Public Administration: Oxymoron or reality? An experience report", CAiSE Industry Track, vol. 1017 de CEUR Workshop Proceedings, págs. 1-8, CEUR-WS.org (2013), en línea en [http://ceur-ws.org/Vol-1017/Paper1CAiSE\\_IT2013.pdf](http://ceur-ws.org/Vol-1017/Paper1CAiSE_IT2013.pdf)

Software Engineering Institute de la CMU. "Contracting for Agile Software Development in the Department of Defense: An Introduction" (agosto 2015), en línea en [resources.sei.cmu.edu/asset\\_files/TechnicalNote/2015\\_004\\_001\\_442515.pdf](resources.sei.cmu.edu/asset_files/TechnicalNote/2015_004_001_442515.pdf)

Government Accountability Office de EE. UU. "Software Development: Effective Practices and Federal Challenges in Applying Agile Methods", GAO-12-681, en línea en <http://www.gao.gov/assets/600/593091.pdf>

Balter B. "Towards a More Agile Government", 41 Pub. Cont. L. J. 149, The Public Contract Law Journal (otoño 2011), en línea en <https://ben.balter.com/2011/11/29/towards-a-more-agile-government/>



Esta obra está sujeta a una licencia de Reconocimiento-CompartirIgual 4.0 Internacional de Creative Commons, disponible en: <https://creativecommons.org/licenses/by-sa/4.0/>