

Universidad de Alcalá



Escuela Politécnica Superior

**Master Universitario en Dirección de Proyectos de
Informáticos**

Trabajo Fin de Master

“Despliegue de productos y servicios”

Autor: D. Ignacio Pérez de Benito

Julio 2020

Universidad de Alcalá

Escuela Politécnica Superior

**Master Universitario en Dirección de Proyectos de
Informáticos**

Trabajo Fin de Master

**“DESPLIEGUE DE PRODUCTOS
Y SERVICIOS”**

Autor : D. Ignacio Pérez de Benito

Director Master : Dr. D. Roberto Barchino Plata

Tribunal evaluador :

Presidente del Tribunal :

Vocal 1º:

Vocal 2º:

Calificación : _____

Alcalá de Henares a, 10 de Julio del 2020

Índice

Resumen.....	6
Abstract.....	6
1 Introducción.....	7
1.1 Definición y alcance.....	8
1.1.1 Propósito.....	8
1.1.2 Objetivo.....	8
1.1.3 Alcance.....	8
1.2 Metodología.....	8
2 Preparando del terreno.....	10
2.1 Recursos y participantes.....	10
2.1.1 Grupos participantes (comunes).....	10
2.1.2 Recursos comunes.....	13
2.2 Asignaciones de grupos participantes a recursos comunes.....	15
2.3 Entornos.....	16
3 Definición del proyecto.....	18
3.1 Entidades de proyecto.....	18
3.2 Condiciones de subida para pasar de un entorno a otro.....	20
3.3 Módulos y servicios expuestos.....	21
3.3.1 Módulos.....	21
3.3.2 Servicios expuestos.....	22
3.4 Diagrama de dependencias.....	23
3.5 Detalles de los diferentes entornos del proyecto.....	26
4 El sprint.....	30
4.1 Procesos del desarrollo por sprints.....	32
4.1.1 Planificar sprint.....	32
4.1.2 Implementar tareas.....	33
4.1.3 Subir a DES.....	34
4.1.4 Realizar pruebas DES.....	34
4.1.5 Solicitar subida PRE.....	35
4.1.6 Subir a PRE.....	35
4.1.7 Realizar pruebas PRE.....	36
4.1.8 Realizar pruebas (cliente).....	36
4.1.9 Solicitar subida PRO.....	37
4.1.10 Subir a PRO.....	38
4.1.11 Esperar visto bueno definitivo.....	38

4.1.12	Consolidar versión	39
4.1.13	Volver a versión anterior.....	39
4.2	Documentos creados	40
5	Subida a los distintos entornos.....	52
5.1	Sobre los despliegues y agrupación de tareas	52
5.2	Tipos de subidas.....	52
5.3	El subproceso de subida a DES/PRE/PRO.....	53
5.3.1	Comprobar precondiciones	54
5.3.2	Coordinar con resto de proyectos	54
5.3.3	Fusionar planes despliegue	55
5.3.4	Asignar fecha y recursos	55
5.3.5	Notificar afectados.....	56
5.3.6	Esperar hasta subida	56
5.3.7	Ejecutar plan de despliegue.....	56
5.3.8	Realizar pruebas básicas	56
5.4	El procedimiento “ejecutar plan de despliegue”.....	58
5.4.1	Notificar comienzo despliegue.....	59
5.4.2	Notificar a responsable de paso siguiente.....	59
5.4.3	Realizar siguiente paso.....	59
5.4.4	Notificar despliegue correcto	60
5.4.5	Subsanar errores no previstos.....	60
5.4.6	Notificar despliegue fallido	61
5.5	El plan de despliegue.....	61
6	Cuadro de mando y mejora continua	64
6.1	Objetivos recomendados.....	64
6.1.1	Reducción del tiempo medio de despliegues.....	64
6.1.2	Reducción de casos en los que el despliegue se desvía de la estimación inicial más allá de un margen razonable.....	66
6.1.3	Documentación actualizada de los servicios expuestos a terceros.....	67
6.2	Ejemplo de cuadro de mando.....	69
7	Conclusiones y valoración personal	70
	Glosario.....	71
	Bibliografía.....	72

Índice de tablas e ilustraciones

Ilustración 1: Ventajas de los procesos bien documentados (Visio). Elaboración propia.....	7
Ilustración 2: Metodología - Fase 1 (Visio). Elaboración propia.....	9
Ilustración 3: Metodología - Fase 2 (Visio). Elaboración propia.....	9
Ilustración 4: Diagrama de grupos participantes. Elaboración propia.....	13
Ilustración 5: Diagrama entorno genérico (Visio). Elaboración propia.....	25
Ilustración 6: Proceso de desarrollo por sprints (Bizagi). Elaboración propia.....	31
Ilustración 7: Frecuencia de despliegues por entorno (Excel). Basado en gráfico de www.if4it.com (7).	52
Ilustración 8: Diagrama de subproceso de subida a DES/PRE/PRO (Bizagi). Elaboración propia.....	53
Ilustración 9: Diagrama de ejecutar plan de despliegue	58
Ilustración 10: Gráfico de líneas de tiempo por despliegue. Elaboración propia.	65
Ilustración 11: Gráfico de líneas de tiempo medio por despliegue hasta sprint n. Elaboración propia.....	65
Ilustración 12: Diagrama de barras de desviación de estimación. Elaboración propia.....	67
Ilustración 13: Porcentaje de documentación. Elaboración propia.....	68

Resumen

Esta guía pretende clarificar dudas que puedan surgir acerca de los despliegues, así como proveer una solución en forma de procesos y procedimientos que aseguren la calidad del producto y la satisfacción del cliente.

Primero se analizará el contexto general, entendiéndose este como la lista de recursos y servicios comunes que sirven como base para cualquier proyecto, así como los equipos de personas encargados de tareas de mantenimiento y despliegues. En segundo lugar, se definirán los proyectos que se quieran incluir en el alcance, y se anotarán las dependencias con recursos y otros proyectos. Desde entonces, todos los desarrollos adoptarán la metodología Scrum y seguirán los diagramas de procesos propuestos.

Para concluir, se sugerirá la elaboración de cuadros de mando y representaciones gráficas de distintos indicadores como parte del plan de mejora continua. De esta manera, la detección de anomalías será más rápida, así como la ejecución de medidas correctivas.

Palabras clave: servicios, proceso, procedimiento, despliegue, entorno; Scrum.

Abstract

This guide is intended to clarify most questions that might arise regarding deployments, as well as providing a solution in form of processes and procedures that ensure product quality and client satisfaction.

Firstly, the general context will be analysed, understanding it as a list of common resources and services used as a basis for any project, and teams in charge of maintenance tasks and deployments. Second, projects which are to be included in the scope will be defined, together with their dependences on resources and other projects. From then, all developments will adopt the Scrum methodology and follow the proposed procedure diagrams.

To conclude, the development of dashboards and graphic representations for the different indicators as part of the continuous improvement plan will be suggested. In this way, anomaly detection will be quicker, and so the execution corrective measures.

Key words: services, process, procedure, deployment, environment, Scrum.

1 Introducción

Las subidas de productos o servicios software a los diferentes entornos, sobre todo al de producción, son uno de los principales quebraderos de cabeza de los proyectos informáticos. Para realizar un despliegue, normalmente se requiere la participación de varios equipos con varias personas que pueden estar o no en la misma ubicación, y que pueden tener mayor o menor conocimiento de las tareas que deben realizar. Cuando no existe un protocolo a seguir, y falta o es insuficiente la documentación asociada, existe un alto porcentaje de probabilidades de que alguien cometa algún error que derive en demoras, no conformidades, y/o cancelaciones de la subida.

La elaboración de un procedimiento de subida conlleva documentar cada paso necesario, describiendo qué hacer, cómo y quién. Es especialmente importante facilitar la comunicación entre los diferentes equipos involucrados. Seguir estas pautas se traducirá inmediatamente en un incremento de la eficiencia y eficacia, así como en el aumento o sostenimiento de la satisfacción del cliente.



Ilustración 1: Ventajas de los procesos bien documentados (Visio). Elaboración propia.

A lo largo de mi experiencia laboral he participado en diversos proyectos en los que se trabajaba de manera diferente; unas veces más ordenada, otras más caóticas, y con distintos resultados. Sin embargo, incluso en los proyectos en los que existía un protocolo de subidas de obligado cumplimiento, ocasionalmente se pasaba por alto para ganar en rapidez cuando alguien te decía “hay prisa”, “es urgente” o “no hay tiempo para eso”. Además, cuando en estos casos el resultado era satisfactorio, se creaba una falsa sensación de seguridad que hacía más probable que se ignoraran de nuevo en futuros despliegues.

El protocolo de subidas viene normalmente impuesto por la empresa (la propia o la del cliente), y no se suele tener mucho poder para modificarlo. Probablemente, esto es lo que me ha decidido a elegir este tema para el TFM. En mi opinión, es una oportunidad para

marcarme un objetivo, un reto: ver hasta qué punto puedo dar una mejor solución a un problema típico en las empresas de desarrollo de software.

1.1 Definición y alcance

1.1.1 Propósito

El propósito de este TFM es el de crear una guía que contenga los pasos y recomendaciones a seguir para subir productos y servicios software a entornos previos y productivos. El nivel de detalle será alto, pero suficientemente flexible como para que pueda usarse en diferentes escenarios.

1.1.2 Objetivo

Al finalizar, debo haber realizado un estudio estructurado y razonado de cómo implementar un procedimiento de subidas de servicios/software a entornos previos y productivos basándome en las materias recibidas durante el máster. Principalmente serán conceptos tomados de la asignatura de Calidad Informática, ya que el tema elegido trata de crear una serie de pautas en forma de procedimientos que puedan ser fácilmente adoptados y entendidos por todas las partes involucradas en una subida. Por tanto, estará perfectamente definido:

- Qué hay que hacer
- Quién ha de hacerlo
- Cómo ha de hacerlo

Además, el estudio debe ser suficientemente amplio como para poder tomarlo como base para una aplicación con la que se pueda gestionar todo lo descrito en el trabajo.

También se incluirá una sección que trate la mejora continua. Se establecerá un sistema de medición y control para evaluar las subidas, así como un conjunto de posibles medidas correctivas a aplicar en los siguientes *sprints*.

1.1.3 Alcance

La presente guía tiene como alcance todos los productos o servicios que formen parte de los proyectos de un departamento de desarrollo de software orientado a la propia empresa o al cliente. Quedan excluidos otros departamentos o programas no desarrollados por la empresa.

Asimismo, las dependencias de otros servicios y recursos serán registrados. Se podrá estudiar así el efecto que tienen los cambios o la no disponibilidad de alguno de estos sobre la aplicación y viceversa.

1.2 Metodología

La primera fase consiste en identificar qué entidades considerar a la hora de realizar el despliegue. Básicamente, es listar los recursos y servicios comunes, y equipos de personas que intervienen, siempre que estén dentro del alcance definido.

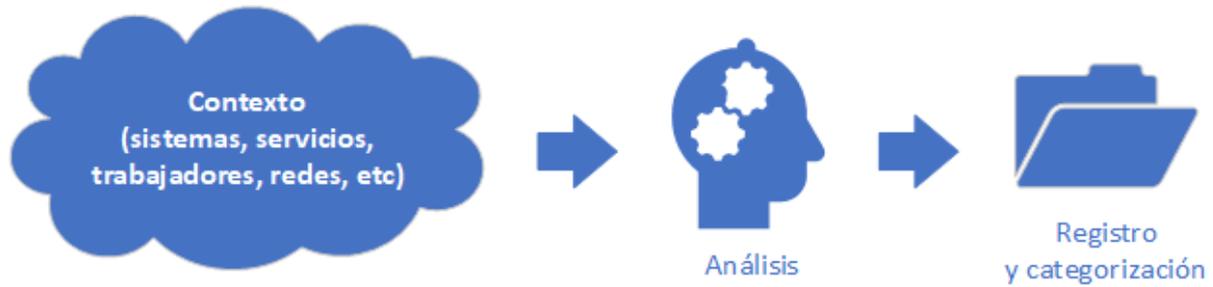


Ilustración 2: Metodología - Fase 1 (Visión). Elaboración propia.

Una vez identificadas las entidades, se procede a registrarlas en diferentes fichas según su naturaleza usando unas plantillas concretas.

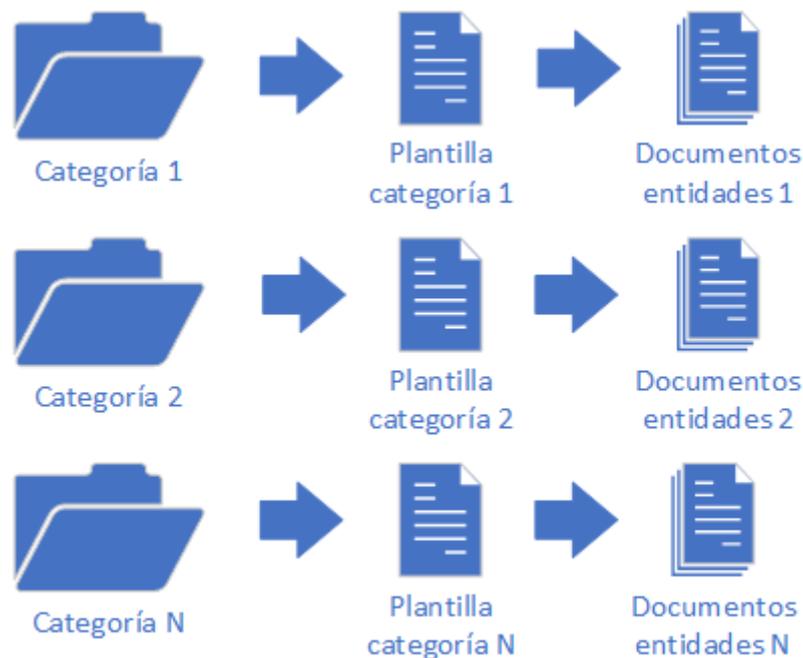


Ilustración 3: Metodología - Fase 2 (Visión). Elaboración propia.

En la segunda, se registran proyectos que vayan a seguir el protocolo de despliegue, así como sus elementos más representativos. Junto con las entidades comunes usadas y otros proyectos de los que dependa, se conforma un grafo de dependencias que ayudará a determinar el impacto de la no disponibilidad de algún elemento.

La tercera describe y representa un *sprint* como un proceso que abarca desde la concepción del mismo hasta su implantación (o marcha atrás). El subproceso de subida a los distintos entornos se tratará en una sección aparte, aunque se considere la misma fase.

Adicionalmente, se incluye un cuadro de mando para realizar un seguimiento, y aplicar la mejora continua.

2 Preparando del terreno

Al igual que un arquitecto empieza una casa por los cimientos y no por el tejado, se deberá documentar primero una serie de elementos comunes. Estos son la base sobre la que se sustentarán los diferentes proyectos a analizar. Además, se deberá decidir quién es responsable de cada recurso común, y acordar una nomenclatura para definir los diferentes entornos.

La información y tablas generadas en los distintos apartados del presente documento se almacenarán preferentemente en un sistema de control de documentos que permita:

- Control de acceso
- Gestión de permisos de escritura/lectura
- Versionado y trazabilidad

En caso de no tener de ninguno, la organización realizará un estudio para determinar el que mejor encaje dentro de las necesidades y el presupuesto disponible. Los productos comerciales más famosos son SharePoint y Confluence. Este último sería especialmente recomendable si se optara por Jira para la gestión de proyectos y tareas. Independientemente de la elección final, la documentación que se vaya generando por la aplicación de los puntos de este apartado se guardará en unas rutas concretas.

Ejemplo de estructura de directorios o secciones:

Documentos	Primer nivel	Segundo nivel	Permisos*	
			Escritura	Lectura
Grupos participantes (comunes)	/resourcesandparticipants	/participantgroups	Sistemas	Dpto. desarrollo
Recursos comunes		/commonresources	Sistemas	Dpto. desarrollo
Asignaciones de grupos participantes a recursos comunes	/assignments		Sistemas	Dpto. desarrollo
Entornos	/environments		Sistemas	Dpto. desarrollo

*Los permisos de acceso no serán concedidos automáticamente. Deberán autorizarse, aunque pertenezcan al grupo.

También podría valorarse la construcción de un software a medida, pero teniendo en cuenta que su implantación sería a largo plazo.

2.1 Recursos y participantes

En este apartado se identifican cuáles son los recursos comunes y potenciales participantes.

2.1.1 Grupos participantes (comunes)

Ruta documentación: /resourcesandparticipants/participantgroups

Se considera grupo participante común a cualquier grupo o puesto de responsabilidad que se encargue de velar por el correcto funcionamiento de los recursos comunes y que, no formando parte de un proyecto, acomete actuaciones solicitadas en despliegues. Pertencerán a una o varias áreas:

- Base de datos
- Sistemas

- Seguridad
- Probadores de software (también llamados *testers*)

Estos grupos están compuestos por personas, y una persona puede pertenecer a varios de estos si procede. La clasificación debe estar íntimamente ligada al organigrama de la empresa.

A continuación, la lista de campos que se necesita registrar de cada uno de ellos:

- Código identificativo único
 - Prefijo propuesto: CGR (Common groups)
 - Formato: CGR-0000
- Nombre descriptivo
 - Debe ser suficientemente aclarativo como para saberse a cuál se refiere
 - Alfanumérico
- Grupo padre
 - Si el presente se trata de un subgrupo, se anota aquí el código del grupo del que depende. En caso contrario, se rellena con un "N/A".
- Correo electrónico
 - Correo electrónico del grupo. Todas las personas que pertenezcan a él deben tener acceso a esta cuenta. No necesariamente todos los que pertenecen a un subgrupo tienen que tener acceso a la cuenta del grupo padre.
- Área
 - Área a la que pertenece el grupo
- Teléfono / Extensión
 - Datos en caso de querer contactar telefónicamente con este grupo
- Calendario
 - Se indican los días y horario de trabajo del grupo
 - Si es un grupo padre, puede obviarse este campo
 - Si es poco práctico o imposible calcularlo con anterioridad, como pueda ser en caso de guardias, se deja vacío
- Observaciones
 - Campo de texto libre para notas adicionales

Ejemplos en plantilla para los grupos de sistemas:

CGR-0001	Grupo sistemas genérico
Grupo padre	N/A
Correo electrónico	sistemas@consultora.es
Área	Sistemas
Teléfono / Extensión	919876541 / 541
Calendario	N/A
Observaciones	A efectos de consultas y peticiones originadas por otras áreas, este es el grupo al que deberán dirigirse. No debe asignarse a ningún recurso.

CGR-000002	Grupo sistemas mañana
Grupo padre	CGR-000001 (Grupo sistemas genérico)
Correo electrónico	sistemas.manana@consultora.es
Área	Sistemas

Teléfono / Extensión	919876542 / 542
Calendario	Calendario laboral Madrid L-V: 7:30-14:30
Observaciones	Este grupo se encarga de las actuaciones y resolución de incidencias en horario de mañana.
CGR-000003	Grupo sistemas tarde
Grupo padre	CGR-000001 (Grupo sistemas genérico)
Correo electrónico	sistemas.tarde@consultora.es
Área	Sistemas
Teléfono / Extensión	919876543 / 543
Calendario	Calendario laboral Madrid L-V: 14:30-21:30
Observaciones	Este grupo se encarga de las actuaciones y resolución de incidencias en horario de tarde.

CGR-000004	Grupo sistemas guardias
Grupo padre	CGR-000001 (Grupo sistemas genérico)
Correo electrónico	sistemas.guardia@consultora.es
Área	Sistemas
Teléfono / Extensión	919876544 / 544
Calendario	Especial
Observaciones	Este grupo se encarga de las actuaciones y resolución de incidencias que no pueden esperar a ser atendidas por los grupos de mañana o tarde.

La realización de un diagrama representando los distintos grupos y sus jerarquías se convierte en la mejor forma de entender su organización:

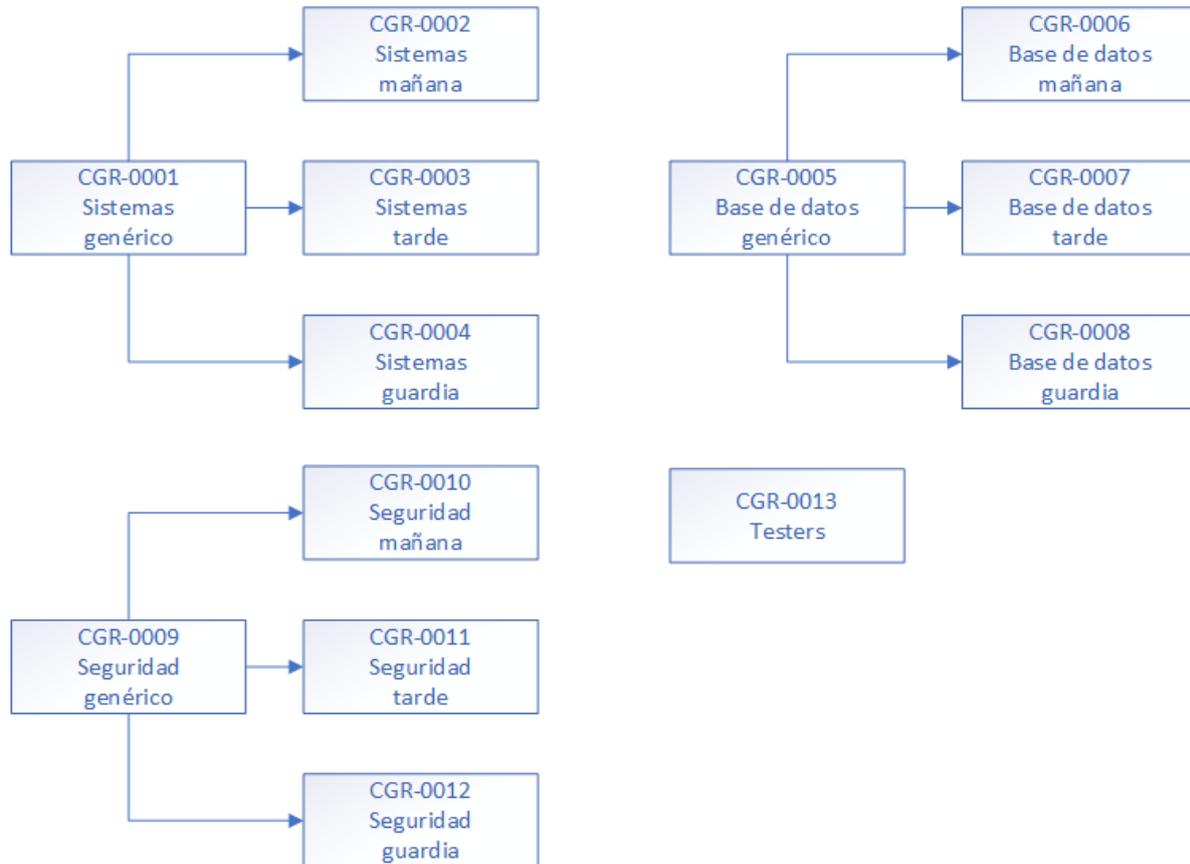


Ilustración 4: Diagrama de grupos participantes. Elaboración propia.

2.1.2 Recursos comunes

Ruta documentación: /resourcesandparticipants/commonresources

Se considera recurso a cualquier elemento necesario para que una aplicación o servicio pueda desplegarse. Para cada uno de estos tendremos que registrar:

- Código identificativo único
 - Prefijo propuesto: CRE
 - Formato: CRE-[Código de tipo]-0000
 - Lista de códigos de tipo posibles:
 - APPSVR (Application Server)
 - Código para los servidores de aplicaciones
 - WEBSVR (Web Server)
 - Código para los servidores web
 - DB (Database)
 - Código para las bases de datos
 - LDAP
 - Código para los servidores de LDAP
 - EXCHFOLDER

- Código para las carpetas de intercambio
- OTHER
 - Código para los recursos que no encajen dentro de los tipos definidos anteriormente
- Nombre descriptivo
 - Debe ser suficientemente aclarativo como para saberse a cuál se refiere
 - Alfanumérico
- Tipo
 - A seleccionar entre estos:
 - Servidor de aplicaciones
 - Servidor HTTP
 - Base de datos
 - LDAP
 - Carpetas de intercambio
 - Otros
 - Nota: El tipo seleccionado determina el código de tipo a la hora de formar el código único
- Producción. Valores posibles:
 - Sí
 - Se trata de un recurso de un entorno de producción
 - No
 - No se trata de un recurso de un entorno de producción
- Máquina/s
 - Lista de nombres de dominio
 - Nota: Se evitará el uso de direcciones IP
- Puertos usados
 - En caso de proceder, se listarán los puertos usados por el recurso
- Dirección del servicio
 - Ruta de acceso al recurso
- Observaciones
 - Cualquier otro dato de interés no recogido en los campos anteriores. Por ejemplo, datos sobre usuarios de acceso y gestión de la máquina o servicio.

Ejemplos en plantilla:

CRE-APPSVR-0001	Apache Tomcat previos 1		
Tipo	Servidor aplicaciones	Producción	No
Máquina/s	servpre01		
Puertos usados	9080		
Dirección	http://servpre01:9080/admin		
Observaciones	Usuario gestión: u3495		

CRE-APPSVR-0002	Apache Tomcat previos 2		
Tipo	Servidor aplicaciones	Producción	No
Máquina/s	servpre02		
Puertos usados	9080		
Dirección	http://servpre02:9080/admin		

Observaciones	Usuario gestión: u3495
----------------------	------------------------

CRE-WEBSVR-0001	Apache Web Server previos 1		
Tipo	Servidor web	Producción	No
Máquina/s	servpre01		
Puertos usados	9090		
Dirección	http://servpre01:9999/ApacheGui		
Observaciones	Usuario gestión: u3495		

CRE-WEBSVR-0002	Apache Web Server previos 2		
Tipo	Servidor web	Producción	No
Máquina/s	servpre02		
Puertos usados	9090		
Dirección	http://servpre02:9999/ApacheGui		
Observaciones	Usuario gestión: u3495		

CRE-DB-0001	Oracle previos 1		
Tipo	Base de datos	Producción	No
Máquina/s	datapre01		
Puertos usados	1521		
Dirección	N/A		
Observaciones	Base de datos Oracle para entornos de PRE		

2.2 Asignaciones de grupos participantes a recursos comunes

Ruta documentación: /assignments

Una vez creadas las anteriores entidades, hay que relacionarlas. Cada recurso ha de asignarse a uno o varios grupos participantes de manera que todo recurso tenga soporte durante las franjas que se consideren oportunas. Gracias a esta asociación también podremos saber con quién se debe contactar en caso de algún problema o petición de actuación sobre los recursos.

Para calcular a qué grupo corresponde el soporte a una determinada hora (si es que estuviera dentro del horario de soporte del recurso) se recomienda el uso de alguna herramienta software, ya que acorta enormemente los tiempos para averiguarlo y reduce los errores.

Se debe usar el teléfono o correo electrónico del grupo padre cuando las comunicaciones sean entre distintas áreas para una mejor atención y gestión.

Ejemplos de asociación:

Recursos	Grupos
CRE-APPSVR-0001 Apache Tomcat previos 1	CGR-000001 Grupo sistemas turno mañana
	CGR-000002 Grupo sistemas turno tarde
CRE-APPSVR-0002 Apache Tomcat previos 2	CGR-000001 Grupo sistemas turno mañana
	CGR-000002 Grupo sistemas turno tarde
CRE-WEBSVR-0001 Apache Web Server previos 1	CGR-000001 Grupo sistemas turno mañana
	CGR-000002 Grupo sistemas turno tarde
CRE-WEBSVR-0002 Apache Web Server previos 2	CGR-000001 Grupo sistemas turno mañana
	CGR-000002 Grupo sistemas turno tarde
...	...

2.3 Entornos

Ruta documentación: /environments

La cantidad de entornos previos que use cada proyecto depende de diversos factores (1) (2) (3):

- De si es un desarrollo a medida o no
 - Si es software *out of the box*, los entornos relacionados con los desarrolladores son prescindibles, siendo suficiente con disponer de uno de preproducción.
- De la complejidad del proyecto
 - Si la lógica de negocio es compleja y/o se necesitan muchas pruebas para verificar el funcionamiento de la aplicación, convendrá tener un entorno específico de pruebas para que los *testers* hagan su trabajo. Por el contrario, en los proyectos simples puede darse el caso de que las pruebas que hagan los desarrolladores sean suficientes y, por tanto, prescindir de él.
- Del tamaño del proyecto
 - Si la cantidad y duración de las pruebas es considerable, se recomienda tener un entorno de pruebas.
- Del presupuesto y personal cualificado del que se disponga
 - Los servidores (y licencias de algunos programas), junto con el coste de personal que garantice el funcionamiento de los mismos puede limitar el número de entornos de los que pueda disponerse, sobre todo si hablamos de pequeñas empresas o proyectos de bajo presupuesto. No obstante, el ahorrar dinero en entornos previos puede resultar más caro a la larga en determinados escenarios.
- De los requisitos propios del proyecto

- El cliente puede imponer que exista un número determinado de entornos previos, más si cabe si se trata de un desarrollo a implementar en sus propias oficinas con su infraestructura.

- Del tiempo que se disponga para acometer los cambios
 - Un número excesivo de entornos previos puede ralentizar que los cambios lleguen a producción. Todo salto entre entornos tiene un coste temporal, independientemente de que haya herramientas que los automaticen total o parcialmente.

La lista de posibles entornos, orden y sus códigos se debe homogeneizar, por lo que se presenta esta tabla como referencia:

Orden	Entorno	Código entorno	Obligatoriedad
1	Local	LOC	N/A
2	Desarrollo	DES	Según proyecto
3	Integración	INT	Según proyecto
4	Pruebas	TEST	Según proyecto
5	Preproducción	PRE	Sí
6	Producción	PRO	Sí

El escenario más frecuente, por experiencia propia, suele ser el de disponer de dos o tres entornos previos: DES, INT (no siempre) y PRE.

3 Definición del proyecto

A la hora de definir un proyecto se elaborará una ficha genérica, unas condiciones de subida, una lista de módulos que lo componen, otra de servicios expuestos, un diagrama de dependencias y unas tablas con los detalles de cada entorno sobre el que se monta la aplicación. Los documentos generados se organizarán de esta manera:

Documentación*	Primer nivel	Permisos**	
		Escritura	Lectura
Entidades de proyectos	/projects	Jefes de proyecto	Dpto. desarrollo
Condiciones de subidas entre entornos	/projects/[nombre corto proyecto]/conditions	Equipo del proyecto	Dpto. desarrollo
Módulos	/projects/[nombre corto proyecto]/modules	Equipo del proyecto	Dpto. desarrollo
Servicios expuestos	/projects/[nombre corto proyecto]/exposedservices	Equipo del proyecto	Dpto. desarrollo
Diagrama de dependencias	/projects/[nombre corto proyecto]/dependencies	Equipo del proyecto	Dpto. desarrollo
Detalles de los diferentes entornos del proyecto	/projects/[nombre corto proyecto]/environments	Equipo del proyecto	Dpto. desarrollo

*Las entidades de proyectos se guardan en el mismo documento, mientras que el resto lo harán en sus respectivas rutas según proyecto. **Los permisos de acceso no serán concedidos automáticamente. Deberán autorizarse aunque pertenezcan al grupo.

La principal dificultad en este apartado es que el diagrama de dependencias y los detalles de los diferentes entornos del proyecto pueden apuntar a elementos de otros proyectos que aún no han sido registrados. Si ocurriese esto, se deberá solicitar la compleción de las fichas de dichos elementos (aunque sea de manera parcial) para así obtener un código y nombre válidos a los que hacer referencia. El objetivo final, no obstante, es que todos los proyectos estén documentados a todos los niveles que indica la presente guía.

Nota aclarativa

El campo “versión y fecha de actualización” de todas las tablas que aparezcan a continuación se referirá al número de *sprint*¹ y la fecha en la que se actualizó por última vez algún dato de la tabla.

3.1 Entidades de proyecto

Ruta documentación: /projects

El proyecto es la entidad más importante en este estudio. Cada vez que se cree uno, se registrarán estos datos:

- Código identificativo único
 - Prefijo propuesto: PRJ

¹ Para más información, consultar en la sección de definición de la tabla de *sprint*.

- Formato: PRJ-000000
- Nombre descriptivo
 - Debe ser suficientemente aclarativo como para saberse a cuál se refiere
 - Alfanumérico
- Nombre corto
 - Código alfanumérico único a usar cuando se tenga que hacer referencia al proyecto en otros documentos o carpetas
 - Máximo 10 caracteres
- Cliente
 - Cliente para el que se realiza el proyecto
 - En caso de ser interno, rellenar con “Interno”
- Fecha de inicio
 - Fecha de comienzo del proyecto
- Fecha fin
 - N/A
 - Desarrollos pendientes y/o mantenimiento
 - Con fecha
 - Ciclo de vida finalizado
- Repositorio de código
 - Dirección absoluta del repositorio de código
- Repositorio de documentación
 - Dirección absoluta del repositorio de documentación
 - Aquí se incluirán los distintos entregables y documentos necesarios para las subidas
- Correo electrónico
 - El habilitado para las comunicaciones relativas al proyecto
 - Es el que ha de usarse para contactar con el equipo de desarrollo
 - Todos los miembros del equipo tendrán acceso a esta cuenta
- Tecnologías/librerías
 - Las usadas por la aplicación
- Responsables del proyecto
 - Jefe de departamento (nivel 2)
 - Nombre
 - Datos de contacto (teléfono, email)
 - Jefe de proyecto (nivel 1)
 - Nombre
 - Datos de contacto (teléfono, email)
- Datos de cliente (si procede)
 - Responsable
 - Nombre
 - Datos de contacto (teléfono, email)
 - Contacto alternativo
 - Nombre
 - Datos de contacto (teléfono, email)
- Equipo de desarrollo del proyecto
 - Cargo
 - Nombre
 - Datos de contacto (teléfono, email)

- Lista de entornos
 - Entornos por los que ha de pasar el servicio o aplicación. Para cada uno:
 - Orden: Determina el orden dentro de la lista de entornos
 - Código de entorno
 - Código del entorno según tabla de entornos.
 - Referencia del entorno
 - Prefijo propuesto: ENV
 - Formato: ENV-[nombre corto proyecto]-[código entorno]

Ejemplo en plantilla:

PRJ-000001	Aplicación de web de subastas para el cliente ACME		
Nombre corto	ACMEMAYHEM	Cliente	ACME
Fecha inicio	01/01/2020	Fecha fin	N/A
Repositorio código	http://mirepo.es/projects/ACMEMAYHEM		
Repositorio documentación	http://myconfluence.es/projects/ACMEMAYHEM		
Correo electrónico	acmemayhem@consultora.es		
Tecnologías / librerías	Java, Hibernate, Angular, ...		
Responsables del proyecto			
Jefe de departamento (nivel 2)	Fulano García	fulano.garcia@consultora.es 654321987	
Jefe de proyecto (nivel 1)	Mengano Santos	mengano.santos@consultora.es 654321988	
Datos de cliente			
Responsable	Joe Smith	joe.smith@acme.com 654123456	
Contacto alternativo	John Wilson	john.wilson@acme.com 654123457	
Equipo de desarrollo del proyecto			
Puesto	Nombre	Contacto	
Analista funcional	Dilbert	dilbert@consultora.es 654321989	
...	
...	
Lista de entornos			
Orden	Código de entorno	Referencia entorno	
1	DES	ENV-ACMEMAYHEM-DES	
2	PRE	ENV-ACMEMAYHEM-PRE	
3	PRO	ENV-ACMEMAYHEM-PRO	

3.2 Condiciones de subida para pasar de un entorno a otro

Ruta documentación: /projects/[nombre corto proyecto]/conditions

Acto seguido, se definen las condiciones para poder pasar de un entorno a otro. Como norma general:

- **Realización de pruebas en entorno anterior.** Antes de subir al siguiente entorno, la aplicación o servicio será probada por los desarrolladores, probadores o cliente. Los

desarrolladores, por norma general, se centrarán más en los primeros entornos. Los probadores, en INT o PRE, y el cliente (uno o varios usuarios), en PRE. Los resultados de las pruebas deben ser satisfactorios.

- **Requiere aprobación.** Subir a preproducción requiere de la aprobación de nivel 1. A producción, tanto de nivel 1 como nivel 2, además del visto bueno del cliente.
- **Compilación.** Debe comprobarse que compila.
- **Pruebas unitarias.** El resultado de todas las pruebas unitarios debe ser satisfactorio.
- **Pruebas básicas.** Una vez desplegada la aplicación, se harán pruebas básicas de navegación entre las distintas secciones, uso de determinadas funcionalidades, comprobación de los resultados, y otros. El conjunto de pruebas a realizar deberá recogerse en un documento. **Importante:** Algunas de las pruebas nunca deberán realizarse en producción si impactan en el negocio. Por ejemplo, no se podrá dar de alta un cliente, modificar un contrato, etc. Se limitarán a operaciones que no cambien el estado de la aplicación.

Ejemplo para proyecto con entornos de DES, PRE y PRO:

PRJ-000001		Condiciones subidas		
Comprobación		LOC→DES	DES→PRE	PRE→PRO
Realización de pruebas en entorno anterior	desarrollador	Obligatorias	Obligatorias	Opcional
	probador	N/A	Opcional	Obligatorias
	cliente	N/A	N/A	Obligatorias
Requiere aprobación		No	Nivel 1	Niveles 1 y 2 Cliente
Compilación		OK	OK	OK
Pruebas unitarias		OK	OK	OK
Pruebas básicas*		OK	OK	OK

*Se describen en otra sección

Algunas de estas tareas pueden y deben ser automatizadas.

3.3 Módulos y servicios expuestos

Dividir la aplicación en módulos independientes es muy recomendable en tanto en cuanto ayuda a la gestión de las dependencias entre los distintos servicios. Así pues, los consumidores de un servicio que se modifica o deja de estar disponible son fácilmente identificables, pudiéndose notificar dicha incidencia o modificación a los responsables de los mismos. Lo mismo ocurre con las carpetas de intercambio.

3.3.1 Módulos

Ruta documentación: /projects/[nombre corto proyecto]/modules

Los módulos de la aplicación se listarán en una misma tabla con los siguientes campos:

- Versión y fecha actualización

- Número de *sprint* y fecha la que se actualizó la tabla por última vez
- Cambios con respecto a última versión
 - No aplica si es la primera versión
 - En las siguientes se indicará qué ha cambiado con respecto a la anterior para así facilitar el trabajo a los que deban adaptar las aplicaciones que usen este servicio.
- Listado módulos con:
 - Nombre corto
 - Descripción

Ejemplo de módulos de una aplicación:

Módulos		
Versión y fecha actualización	1.0	01/01/2020
Cambios con respecto a última versión	N/A (primera versión)	
Nombre corto	Descripción	
BACK	Lógica de negocio	
REPORTS	Módulo que crea informes a partir de los datos	
FRONT	Gestión de la capa de vista	

3.3.2 Servicios expuestos

Ruta documentación: /projects/[nombre corto proyecto]/exposedservices

Los servicios expuestos a terceros se tendrán que registrar los siguientes campos:

- Código identificativo único
 - Prefijo propuesto: SER (Service)
 - Formato: SER-[Nombre corto proyecto]-[Código servicio]
- Nombre descriptivo
 - Debe ser suficientemente aclarativo como para saberse a cuál se refiere
 - Alfanumérico
- Tipo
 - Servicio REST
 - Servicio web
 - Otros
- Versión y fecha actualización
 - Número de *sprint* y fecha la que se actualizó la tabla por última vez
- Proyecto
 - Código del proyecto
- Módulo
 - Módulo de la aplicación que expone el servicio
- Descripción
 - Descripción larga del servicio, pero sin entrar en detalles (para eso está el enlace de la documentación)
- Enlace documentación
 - Debe apuntar a un sitio web o archivo donde se detalle cómo usar el servicio
- Cambios con respecto a última versión
 - No aplica si es la primera versión

- En las siguientes se indicará qué ha cambiado con respecto a la anterior versión

Importante: La documentación sobre cómo usar el servicio debe describir el tipo de operación, los parámetros de entrada y salida, y un ejemplo de llamada. Además, lo escrito sobre versiones anteriores debe ser accesible también.

Ejemplo en plantilla:

SER-ACMEMAYHEM-001		Pedido cliente		
Tipo	Servicio REST	Versión y fecha actualización	1.0	01/01/2020
Proyecto	PRJ-000001	Módulo	informes	
Descripción	Alta de un pedido de un cliente			
Enlace documentación	http://confluence.acme.com/acmemayhem/servicioexternos#pedidocliente			
Cambios con respecto a última versión	N/A (primera versión)			

3.4 Diagrama de dependencias

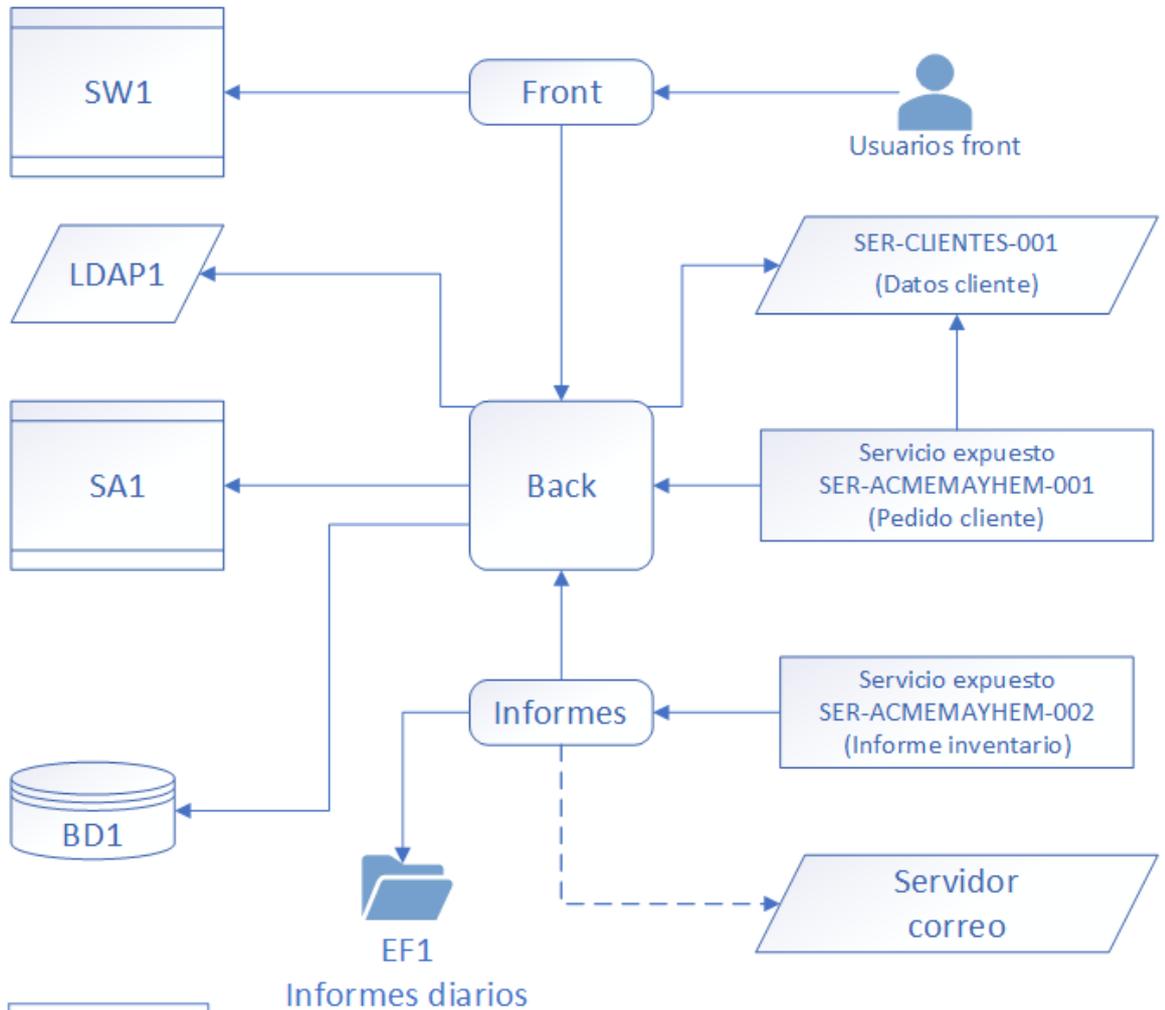
Ruta documentación: /projects/[nombre corto proyecto]/dependencias

El diagrama de dependencias se tendrá como herramienta visual para calcular a qué afecta en la aplicación la no disponibilidad o mal funcionamiento de servidores o servicios de terceros.

Las reglas a seguir en el diagrama son las siguientes:

- Las flechas apuntan al elemento del que dependen. Si el trazo es discontinuo, significa que el elemento origen de la flecha puede operar parcialmente aun no estando disponible el elemento al que apunta.
- Los servidores de aplicaciones que desplieguen módulos idénticos estarán unificados en una sola caja. Aplicable igualmente para los de web.
- Deberán recogerse todos los módulos de la aplicación.
- Tanto los servicios expuestos a terceros como los consumidos externos a la aplicación se asociarán a los módulos que correspondan.
- Si un servicio expuesto a terceros depende de un servicio de terceros, se debe añadir una flecha entre ellos que indique dicha dependencia.
- Los diferentes elementos se representan con nombres genéricos. Los detalles sobre a qué recurso concreto se refieren se listan en las definiciones de cada entorno (previos y producción).
- Los usuarios son elementos especiales en el sentido que conviene notificarles (a todos o parte de ellos) la caída de los elementos que usen. Por ejemplo, una web. Además, lo normal es que los usuarios a notificar sean distintos según el entorno.

Ejemplo simplificado de diagrama:



Leyenda			
	Servidores		Base de datos
	Servicios expuestos a terceros		Recursos comunes o servicios consumidos de terceros
	Usuarios de web o servicios		Dependencia total
			Dependencia parcial
	Carpeta intercambio		Módulos aplicación

Ilustración 5: Diagrama entorno genérico (Visio). Elaboración propia.

3.5 Detalles de los diferentes entornos del proyecto

Ruta documentación: /projects/[nombre corto proyecto]/environments

Esta sección no tratará el entorno local, puesto que no es único y depende de cada desarrollador. No obstante, aunque no se vaya a analizar en esta guía, se recomienda que al menos haya un acuerdo entre los desarrolladores para usar las mismas herramientas y configuración. La creación de un manual de instalación en entornos locales minimizará la aparición de errores, así como el tiempo que necesita un nuevo desarrollador para empezar a ser productivo.

Para cada entorno definido (excepto local) se tendrá una ficha con las siguientes secciones y campos:

Sección principal

- Código identificativo único
 - Prefijo propuesto: ENV
 - Formato: ENV-[Nombre corto proyecto]-[Código entorno]
- Nombre descriptivo
 - Debe ser suficientemente aclarativo como para saberse a cuál se refiere
 - Alfanumérico
- Entorno
 - Nombre largo del entorno que se haya usado en el código identificativo único
- Versión y fecha actualización
 - Número de *sprint* y fecha la que se actualizó la tabla por última vez
- Cambios con respecto a última versión
 - No aplica si es la primera versión
 - En las siguientes se indicará qué ha cambiado con respecto a la anterior

Horario comprometido de funcionamiento

- Calendario
 - Nombre del calendario usado como referencia
- Periodo
 - Para hacer distinción entre horario de verano e invierno
- Días
 - Días de la semana en los que aplica el horario
- Horas
 - Franja horaria de funcionamiento comprometido

Recursos comunes

- Lista de recursos comunes usados indicando su ID.
 - ID diagrama: Identificador del elemento en el diagrama de dependencias
 - ID: Identificador del recurso
 - Observaciones: Campo libre para anotar datos de interés

Base de datos

- ID de la base de datos
 - Lista de SID o nombre de servicios, esquemas y permisos usados

Servicios externos de terceros usados

- ID del servicio o recurso
- Entorno
 - Se indica el entorno para los casos en los que el servicio o recurso esté registrado como proyecto. Para el resto, se indica como “sin determinar”.
- Horarios uso y criticidad

Carpetas de intercambio

- ID diagrama
 - Identificador del elemento en el diagrama de dependencias
- Máquina
 - Máquina en la que se encuentra la carpeta
- Ruta
 - Ruta de acceso a la carpeta
- Permisos
 - Pueden ser de lectura, escritura, ejecución, o combinación de estos
- Usuario
 - Usuario que realiza la operación sobre el directorio

Notificaciones a usuarios

- Lista de usuarios a notificar en caso de no disponibilidad del elemento al que están asociados. Contendrá:
 - ID diagrama
 - Identificador del usuario en el diagrama de dependencias
 - Asunto correo electrónico
 - Asunto de correo electrónico propuesto en caso de notificación
 - Destinatarios
 - Correos electrónicos a los que se enviará la notificación en caso de ser necesaria

Además de esta tabla, será necesario añadir un diagrama de dependencias de los recursos usados por el entorno, de manera que sea sencillo identificar a qué elementos impacta la no disponibilidad de alguno de ellos.

Ejemplo en plantilla:

ENV-ACMEMAYHEM-PRE		Entorno preproducción para ACMEMAYHEM		
Entorno	Preproducción	Versión y fecha actualización	1.0	01/01/2020
Cambios con respecto a última versión	N/A (primera versión)			
Horario comprometido de funcionamiento				
Calendario	Periodo	Días	Horario	
Oficina Madrid	Horario invierno (octubre-mayo)	L-J	8:00 - 19:00	
		V	8:00 - 15:00	
		S,D	N/A	
		Festivos	N/A	
	Horario verano (junio-septiembre)	L-V	8:00 - 15:00	
		S,D	N/A	

		Festivos	N/A	
Recursos comunes				
ID diagrama	ID	Observaciones		
SA1	CRE-APPSVR-0001 (Servidor aplicaciones - servpre01)	En clúster. El elemento sólo se considerará no disponible si ambos lo están al mismo tiempo.		
	CRE-APPSVR-0002 (Servidor aplicaciones - servpre02)			
SW1	CRE-WEBSVR-0001 (Servidor web - servpre01)	En clúster. El elemento sólo se considerará no disponible si ambos lo están al mismo tiempo.		
	CRE-WEBSVR-0002 (Servidor web - servpre02)			
LDAP1	CRE-LDAP-0001 (LDAP)	LDAP usado para autenticación de usuarios en entornos previos		
Servidor correo	CRE-OTH-0001 (Servidor correo)	Servidor de correo usado para el envío de informes.		
Base de datos				
ID diagrama	ID	SID / Nombre servicio	Esquema	Permisos
BD1	CRE-DB-0001	dbservicedes1	mayhem	lectura/escritura
		dbservicedes1	mayhemread	lectura
Servicios externos de terceros usados				
ID / ID diagrama		Entorno	Horario uso	Criticidad
SER-CLIENTES-001		Preproducción	Completo	Alta
Carpetas de intercambio				
ID diagrama	ID	Usuario	Permisos	
SF1	CRE-EXCHFOLDER-0001	mayhem	escritura	
Notificaciones a usuarios				
ID diagrama	Asunto correo electrónico	Destinatarios		
Usuarios <i>front</i>	Indisponibilidad temporal de <i>front</i> de ACMEMAYHEM en entorno de preproducción	<ul style="list-style-type: none"> • usuarios@acme.com • probadores@consultora.es 		

De esta manera, se consigue vincular todos los elementos del gráfico de dependencias con los recursos específicos y saber a qué afecta cada elemento. Caben destacar estos puntos.

- El horario comprometido de funcionamiento sirve principalmente para que el resto de servicios y usuarios que usen la aplicación sepan cuándo es esperable que funcione, y para que cuando un recurso usado por la aplicación tenga que ser desactivado o

parado (por mantenimiento u otra operación que lo exija), no se realice en el horario de uso (si fuera posible).

- Para los casos de *clustering*, un mismo elemento del diagrama tendrá dos o más recursos específicos. Sólo la caída de todos ellos provocará una no disponibilidad.
- Las bases de datos utilizadas incluyen el SID o nombre de servicio, esquema y permisos. Así, si el grupo de bases de datos tuviera que realizar una operación sobre la misma, podrá valorar si afecta o no a la aplicación.
- Para los servicios externos de terceros usados (de otros proyectos) se necesita la columna de entorno ya que, si no, no sería una referencia única. Dentro de la lista de entornos posibles para el servicio, la selección de uno u otro puede depender de:
 - Si es entorno productivo o no. Una aplicación en producción siempre apuntará otros entornos de producción.
 - La disponibilidad y estabilidad. Puede resultar interesante que una aplicación apunte a servicios de entornos previos más cercanos al de producción, siempre y cuando no provoquen comportamientos anómalos en terceros. Los entornos inferiores suelen ser más inestables y soportan más cambios.
- Conviene tener un registradas las carpetas de intercambio. Un cambio en estas puede afectar a la propia aplicación o a terceros.

4 El sprint

Se denomina *sprint* a cada iteración de desarrollo dentro de un proyecto Scrum. Dependiendo de la naturaleza del mismo, puede convenir que sean periodos cortos (más frecuentes) o periodos largos (menos frecuentes). Su duración debe estar comprendida entre dos y cuatro semanas, y mantenerse la misma² para todos los siguientes con el fin de facilitar las mediciones de velocidad del equipo. (4)

Los documentos relevantes de este apartado son los siguientes:

Documentación*	Primer nivel	Permisos**	
		Escritura	Lectura
Definición de sprints	/projects/[nombre corto proyecto]/sprints	Equipo del proyecto	Dpto. desarrollo
Tareas del sprint	/projects/[nombre corto proyecto]/sprints/[nº sprint]/tasks	Equipo del proyecto	Dpto. desarrollo
Descripción de las pruebas	/projects/[nombre corto proyecto]/testsdefinition	Equipo del proyecto	Dpto. desarrollo
Pruebas básicas*	/projects/[nombre corto proyecto]/sprints/[nº sprint]/tests	Equipo del proyecto	Dpto. desarrollo
Pruebas exhaustivas*	/projects/[nombre corto proyecto]/sprints/[nº sprint]/tests	Equipo del proyecto	Dpto. desarrollo
Plan de despliegue	/projects/[nombre corto proyecto]/sprints/[nº sprint]/deploy	Equipo del proyecto	Dpto. desarrollo
No conformidad	/projects/[nombre corto proyecto]/ncr	Equipo del proyecto	Dpto. desarrollo
Plan de marcha atrás	/projects/[nombre corto proyecto]/rollbackplan	Grupo sistemas	Dpto. desarrollo

*Las pruebas básicas y exhaustivas tienen el mismo formato y sólo se diferencian en los test elegidos.

Los hotfix: un caso particular

Los *hotfix* se tratarán como *sprints* a efectos prácticos, pero no se tomarán como referencia para calcular la velocidad al estar compuestos por una o muy pocas tareas. Es importante tener en cuenta que pueden coincidir en tiempo un *sprint* y un *hotfix*, y que en cuanto el *hotfix* se finalice, habrá que integrarlo en el código del *sprint* en curso (cambios en documentación inclusive).

² Salvo razón justificada.

Representación del sprint en un diagrama de procesos (con tres entornos)

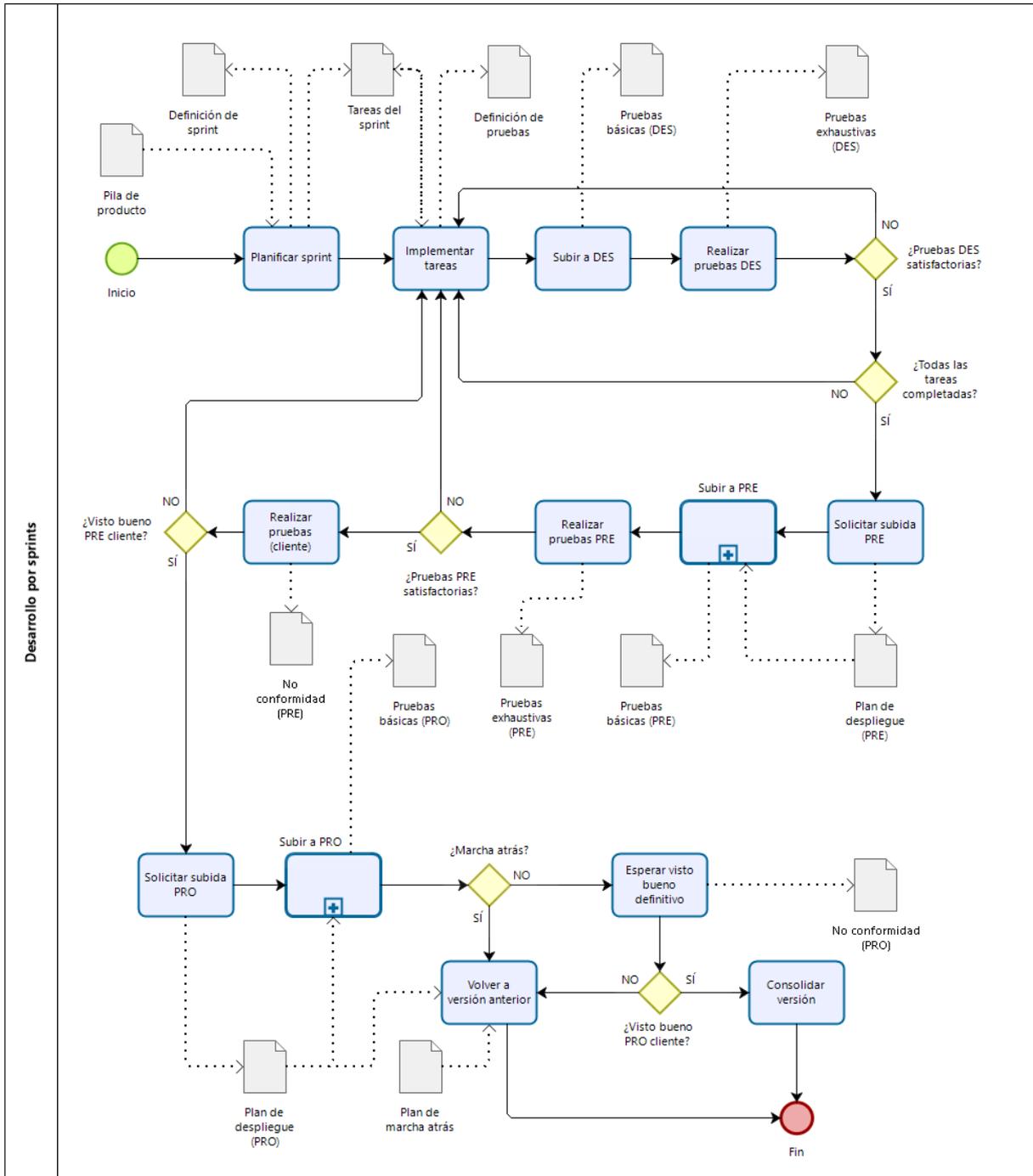


Ilustración 6: Proceso de desarrollo por sprints (Bizagi). Elaboración propia.

Notas:

- La pila de producto se representa en el diagrama como documento por ser la fuente de donde se extraen tareas para el *sprint*, pero en la práctica no se guardará más que en el software de tablero Scrum del proyecto.

- La revisión del *sprint* y la retrospectiva quedan fuera del diagrama por ser eventos realizables en paralelo al flujo.

A continuación, se hará un breve resumen de cada proceso. El objetivo principal, no obstante, será el de detallar el plan de despliegue.

4.1 Procesos del desarrollo por sprints

4.1.1 Planificar sprint

4.1.1.1 Descripción

A la hora de planificar proyectos software, la división del trabajo en tareas es esencial para una buena gestión. En el caso de usar metodologías como Scrum (o similares), estas deben tener un tamaño adecuado para facilitar tanto la monitorización diaria del avance como para identificar riesgos y problemas sin recurrir a procesos de gestión complejos. (4)

En la fase de planificación del *sprint* (o equivalente en caso de no seguir Scrum), las tareas deben estar ya detalladas suficientemente como para saber si modifican o eliminan algún servicio expuesto:

- En caso afirmativo
 - Se identifican los proyectos a los que afectan los cambios y se contacta con los responsables de estos para coordinar los desarrollos necesarios y pactar las fechas de implementación y subida. Si no hubiese ninguna referencia a otros proyectos, pasa a tratarse como en los casos negativos.
 - Si es posible, agrupar las tareas que afectan a terceros (las que modifican servicios externos) e incluirlas en un *sprint* aparte. Igualmente se hará en los proyectos afectados. Así, en caso de algún fallo inesperado al desplegar, las operaciones de vuelta atrás se simplifican y no se causan tantos estragos (ni la pérdida de una larga lista de tareas comprometidas con el cliente). Si no es posible, a la hora de desarrollar se debe empezar por las tareas que afecten a terceros.
 - Tanto el *sprint* que modifica servicios externos de este proyecto como los proyectos asociados que los consumen, deberán compartir un código único especial en sus fichas para indicar este hecho.
- En caso negativo
 - El equipo puede gestionarse como considere oportuno, aunque siguiendo siempre unas recomendaciones.

La manera que se tiene para identificar los proyectos que consumen un servicio expuesto no es más que buscando la referencia del servicio en el resto de fichas de detalle de entorno de todos los proyectos registrados. Nótese que la iniciativa en la gestión del *sprint* recaerá principalmente en el jefe de proyecto de la aplicación que cambia el servicio expuesto.

4.1.1.2 Responsables

- Propietario del producto
- Equipo de desarrollo del proyecto
- Scrum master
- Interesados

4.1.1.3 Documentos

Tipo	Documentos
Entrada	Pila de producto

Entrada/salida	Tabla de resumen de pruebas por entorno
Salida	Definición del sprint Tareas del sprint

4.1.1.4 *Siguiente paso*

- Implementar tareas

4.1.2 Implementar tareas

4.1.2.1 *Descripción*

Los desarrolladores se irán asignando tareas de la pila de *sprint*. Cuando acaban una tarea, deben revisar y completar la ficha de la misma. Acto seguido, se modifican las siguientes fichas si así fuera necesario:

Módulos

Se añadirán o eliminarán módulos de la tabla de módulos si procede.

Servicios expuestos

- Si se añade uno nuevo, crear una nueva ficha rellenando todos los campos pertinentes
- Si es una eliminación, eliminar la ficha
- Si es una modificación, “Cambios con respecto a última versión” tendrá un resumen de los cambios

En cualquiera de los casos, el documento al que se refiera “enlace documentación” deberá ser actualizado. Los cambios definitivos se confirmarán al finalizar el *sprint*.

Diagrama de dependencias

Habrá que modificarlo sí se añaden o eliminan:

- módulos
- servicios expuestos
- servicios consumidos
- recursos comunes
- usuarios
- carpetas de intercambio

O bien, si se modifica alguna de las dependencias entre los elementos o su tipo (dependencia total o parcial).

Tabla de resumen de pruebas por entorno

Modificar en caso de que la tarea necesite de pruebas no contempladas o deje sin efecto alguna de ellas.

Definición de las pruebas

Modificar en caso de que la tarea necesite de pruebas no contempladas, deje sin efecto alguna de ellas o afecte a alguna ya existente.

4.1.2.2 Responsables

- Equipo de desarrollo del proyecto

4.1.2.3 Documentos

Tipo	Documentos
Entrada	Tareas del sprint
Entrada/salida	Tabla resumen de las pruebas por entorno Definición de pruebas Módulos Servicios expuestos Diagrama de dependencias

4.1.2.4 Siguiendo paso

- Subir a DES

4.1.3 Subir a DES

4.1.3.1 Descripción

Este paso consiste en pasar los cambios realizados (las tareas del *sprint* o *bugs* asociados a las mismas) al entorno de desarrollo. A diferencia de los entornos de preproducción y producción, es el equipo de desarrollo el que se encargará de realizarlo.

Nota: Ver sección de “Subida a los distintos entornos” para más información.

4.1.3.2 Responsables

- Equipo de desarrollo del proyecto

4.1.3.3 Documentos

Tipo	Documentos
Salida	Pruebas básicas (DES)

4.1.3.4 Siguiendo paso

- Realizar pruebas DES

4.1.4 Realizar pruebas DES

4.1.4.1 Descripción

El propio equipo de desarrollo realizará las pruebas exhaustivas. Es posible que en algunos casos no se lleven a cabo todas porque esté prevista otra subida a DES en breve, pero antes de una subida a PRE será necesario haber probado a fondo la *build* que integra los cambios a promocionar.

4.1.4.2 Responsables

- Equipo de desarrollo del proyecto

4.1.4.3 Documentos

Tipo	Documentos
Salida	Pruebas exhaustivas (DES)

4.1.4.4 *Siguiente paso*

¿Pruebas DES satisfactorias?	¿Todas las tareas completadas?	Siguiente paso
No	-	Implementar tareas
Sí	No	
Sí	Sí	Solicitar subida PRE

4.1.5 Solicitar subida PRE

4.1.5.1 *Descripción*

Cuando esté suficientemente probada la aplicación en DES con todas las tareas del *sprint* incluidas, se podrá solicitar la promoción a PRE. La petición se realizará a través de la herramienta de *ticketing* implantada en la empresa. Será imprescindible adjuntar el plan de despliegue, así como contar con la aprobación correspondiente (nivel 1).

Los campos mínimos que deben rellenarse en la solicitud son:

- Proyecto
- Código del sprint
- Código del sprint conjunto (si procede)
- Fecha
 - Lo antes posible
 - Fecha concreta
- Hora
 - Indiferente
 - Franja horaria
 - Hora exacta

4.1.5.2 *Responsables*

- Equipo de desarrollo del proyecto

4.1.5.3 *Documentos*

Tipo	Documentos
Salida	Plan de despliegue (PRE)

4.1.5.4 *Siguiente paso*

- Subir a PRE

4.1.6 Subir a PRE

4.1.6.1 *Descripción*

Este paso consiste en pasar los cambios realizados (las tareas del *sprint* o *bugs* asociados a las mismas) al entorno de preproducción.

Nota: Ver sección de “Subida a los distintos entornos” para más información.

4.1.6.2 Responsables

Dependiendo del alcance de los cambios, pueden ser prescindibles algunos, pero por lo general:

- Equipo de desarrollo del proyecto (si se responsabiliza de las pruebas básicas)
- Grupo de sistemas
- Grupo de base de datos
- Grupo de seguridad

4.1.6.3 Documentos

Tipo	Documentos
Entrada	Plan de despliegue (PRE)
Salida	Pruebas básicas (PRE)

4.1.6.4 Siguiete paso

- Realizar pruebas PRE

4.1.7 Realizar pruebas PRE

4.1.7.1 Descripción

La realización de pruebas exhaustivas por parte del equipo de pruebas es imprescindible, ya que en el próximo paso será el cliente el que pruebe la aplicación. Sobre todo, no deben aparecer errores no controlados, ya que son los que más daño pueden hacer a la reputación de la empresa.

4.1.7.2 Responsables

- Equipo de desarrollo del proyecto (opcional)
- Equipo de probadores

4.1.7.3 Documentos

Tipo	Documentos
Salida	Pruebas exhaustivas (PRE)

4.1.7.4 Siguiete paso

¿Pruebas PRE satisfactorias?	Siguiete paso
No	Implementar tareas
Sí	Realizar pruebas (cliente)

4.1.8 Realizar pruebas (cliente)

4.1.8.1 Descripción

Los probadores del cliente realizarán las pruebas que ellos consideren con el fin de validar la versión. Cuando detecten algún error o no estén de acuerdo con alguna de las soluciones aportadas, informarán al equipo del proyecto a través del canal que se considere más oportuno (intentando que sea siempre el mismo).

En este paso es especialmente importante la comunicación con el cliente, ya que podría ser la antesala de una no conformidad. Hay que recordar que el pilar sobre el que se basa la

norma ISO 9001 es la satisfacción del cliente (5). Como norma general, aplicaremos las siguientes medidas:

¿Presente en versión anterior?	¿Tiempo suficiente?	Importancia	Estrategia
No	Sí	-	Registrar la incidencia. Añadir comentario en la ficha de la tarea acerca de esta incidencia. Enlazarla a la incidencia. Resolverla en este mismo sprint.
No	No	-	Añadir comentario a tareas implicadas y devolverlas a la pila de producto. Desplegar en entornos previos sin estos cambios.
Sí	Sí	Crítica	Registrar la incidencia. Añadir tarea al sprint actual*.
Sí	No	Crítica	Registrar la incidencia. Se creará un hotfix para su resolución.
Sí	-	No crítica	Registrar la incidencia.

*En caso de tener dudas sobre si se llega o no a la entrega, tratarla como sin tiempo suficiente.

4.1.8.2 Responsables

- Cliente

4.1.8.3 Documentos

Tipo	Documentos
Salida	No conformidad (PRE)

4.1.8.4 Siguiete paso

¿Visto bueno PRE cliente?	Siguiete paso
No	Implementar tareas
Sí	Solicitar subida PRO

4.1.9 Solicitar subida PRO

4.1.9.1 Descripción

A partir de que el cliente haya dado el visto bueno a los cambios del *sprint* actual, se podrá solicitar promocionar a PRO. La petición se realizará a través de la herramienta de *ticketing* implantada en la empresa. Será imprescindible adjuntar el plan de despliegue, y contar con la aprobación correspondiente (niveles 1 y 2).

4.1.9.2 Responsables

- Equipo de desarrollo del proyecto

4.1.9.3 Documentos

Tipo	Documentos
Salida	Plan de despliegue (PRE)

4.1.9.4 *Siguiente paso*

- Subir a PRO

4.1.10 Subir a PRO

4.1.10.1 *Descripción*

Este paso consiste en pasar los cambios realizados (las tareas del *sprint* o *bugs* asociados a las mismas) al entorno de producción.

Nota: Ver sección de “Subida a los distintos entornos” para más información.

4.1.10.2 *Responsables*

Dependiendo del alcance de los cambios, pueden ser prescindibles algunos, pero por lo general:

- Equipo de desarrollo del proyecto (si se responsabiliza de las pruebas básicas)
- Grupo de sistemas
- Grupo de base de datos
- Grupo de seguridad

4.1.10.3 *Documentos*

Tipo	Documentos
Entrada	Plan de despliegue (PRO)
Salida	Pruebas básicas (PRO)

4.1.10.4 *Siguiente paso*

¿Marcha atrás?	Siguiente paso
No	Esperar visto bueno definitivo
Sí	Volver a versión anterior

4.1.11 Esperar visto bueno definitivo

4.1.11.1 *Descripción*

Se trata de un estado de espera hasta que el cliente dé el visto bueno definitivo. Como normalmente no se pueden hacer pruebas exhaustivas en PRO, se suele dejar una ventana temporal para comprobar que todo funciona como debe. Con esperar dos o tres horas desde que los usuarios empiezan a utilizar la nueva versión suele ser suficiente. Si en ese periodo de tiempo no ha habido ninguna incidencia relevante, se solicitará expresamente el visto bueno al cliente.

4.1.11.2 *Responsables*

- Equipo de desarrollo del proyecto (esperando)
- Cliente (informando de posibles incidentes)

4.1.11.3 *Documentos*

Tipo	Documentos
Salida	No conformidad (PRO)

4.1.11.4 *Siguiente paso*

¿Visto bueno PRO cliente?	Siguiente paso
No	Volver a versión anterior

Sí	Consolidar versión
----	--------------------

4.1.12 Consolidar versión

4.1.12.1 Descripción

Una vez dado el visto bueno definitivo, se realizará la actualización o confirmación de los documentos del *sprint*.

4.1.12.2 Responsables

- Equipo de desarrollo del proyecto

4.1.12.3 Documentos

N/A

4.1.12.4 Siguiete paso

- [Fin]

4.1.13 Volver a versión anterior

4.1.13.1 Descripción

La perfección no existe. Subir código o configuración con errores, o tener algún imprevisto durante el despliegue, es algo de lo que hay que protegerse. En caso de detectar un error durante el proceso (ya sea en el despliegue en sí o durante las pruebas), se notificará el problema a todos los implicados, puesto que se está generando un bloqueo de:

- Recursos técnicos, ya que el software no estará disponible durante más tiempo del planificado
- Recursos humanos, ya que pasos programados para cierta hora se verán retrasados o cancelados, y porque pueden necesitarse otras acciones no previstas

4.1.13.2 Responsables

- Equipo de desarrollo del proyecto (si se responsabiliza de las pruebas básicas)
- Grupo de sistemas
- Grupo de base de datos
- Grupo de seguridad

4.1.13.3 Documentos

Tipo	Documentos
Entrada	Plan de marcha atrás

4.1.13.4 Siguiete paso

- [Fin]

4.2 Documentos creados

Cuando haya que modificarse algún documento de fichas no almacenados en la carpeta del *sprint* específico, no se hará directamente puesto que, al ser genéricos para cualquier entorno, podemos estar variando la foto real de lo que hay en producción. Una aproximación puede ser la de hacer una copia del documento añadiéndole un sufijo al nombre del fichero (por ejemplo, el número del *sprint*), trabajar con esta y fusionarla con el documento definitivo al terminar el *sprint*. La gestión de las versiones transitorias es el punto más complejo de la guía si no se dispone de un software a medida.

Nota: Los permisos sobre estos documentos serán los mismos que los originales.

4.2.1.1 Definición del *sprint*

Ruta documentación: /projects/[nombre corto proyecto]/sprints

Para formalizar un *sprint* se debe rellenar una ficha por cada proyecto vinculado al cambio. Los campos son los siguientes:

- Código identificativo único
 - Prefijo propuesto: SPR
 - Formato: SPR-[Nombre corto proyecto]-0000
- Nombre descriptivo
 - Debe ser suficientemente aclarativo como para saberse a cuál se refiere
 - Alfanumérico
- Número de *sprint*
 - Único
 - Equivale a la versión (ver inicio apartado 6 para más información)
- Tipo *sprint*
 - Estándar: Si es un *sprint*
 - Hotfix: Si es un *hotfix*
- *Sprint* conjunto (S/N)
 - Marcar S o N según sea o no un *sprint* conjunto
- Código *sprint* conjunto único
 - Formato: SPR-SHARED-0000
 - Los *sprints* asociados de otros proyectos deben tener el mismo código
- Fecha prevista inicio
 - Fecha prevista en la que se iniciarán las tareas
- Fecha prevista fin
 - Fecha prevista de finalización de las tareas
- Enlace detalles
 - Dirección dentro del repositorio de documentación en la que se encuentra la información detallada del *sprint*
- N°
 - Número de la tarea dentro del *sprint*
- Tarea
 - Descripción breve de la tarea para que sea fácil su identificación
- Esfuerzo estimado
 - Valor numérico del esfuerzo estimado

- La magnitud depende de la establecida en el proyecto, aunque se recomienda usar el *planning poker* para las estimaciones
- Modifica servicios expuestos (S/N)
 - Para identificar las tareas que modifican servicios expuestos
- Depende de
 - Número de tarea de la que depende. Útil a la hora de planificar, ya que hay casos en los que no se puede comenzar una si no está finalizada otra.
 - “N/A” si no depende de ninguna
- Total (de esfuerzo estimado)
 - Suma del esfuerzo de todas las tareas
- Resumen modifica servicios expuestos (S/N)
 - Valdrá “S” en el momento que alguna de las tareas sea “S”, y “N” en caso contrario

Ejemplo en plantilla:

SPR-ACMEMAYHEM-0001		Sprint 1		
Número de sprint		1.0	Tipo sprint	Estándar <input checked="" type="checkbox"/> / Hotfix <input type="checkbox"/>
Sprint conjunto (S/N)		N	Código sprint conjunto	N/A
Fecha prevista inicio		01/01/2020	Fecha prevista fin	21/01/2020
Enlace detalles		http://myconfluence.es/ACMEMAYHEM/sprint/1.0/		
Tareas				
Nº	Tarea	Esfuerzo estimado	Modifica servicios expuestos (S/N)	Depende de
1	Creación de ventana de gestión de usuarios	5	N	N/A
2	Cambio logo empresa	1	N	N/A
...	
Total		34	N	

4.2.1.2 Definición de tareas

Ruta documentación: /projects/[nombre corto proyecto]/sprints/[nº sprint]/tasks

Para cada una de estas se debe crear una ficha detallada. Los campos que se deben incluir son los siguientes:

- Nombre tarea
 - Nombre de la tarea
- Pertenece a
 - Código del *sprint* al que pertenece
- Número tarea
 - Número que se ha asignado a la tarea dentro del *sprint*
- Scripts
 - Independientes del entorno
 - Nombre y ruta de los scripts asociados a la tarea que se pueden ejecutarse en cualquiera de los entornos

- La columna de “scripts de marcha atrás”, ídem, pero para revertir los cambios del script de su misma fila.
 - Dependientes del entorno
 - Ruta de los scripts asociados a la tarea de los que debe existir una versión por cada entorno
 - La columna de “scripts de marcha atrás”, ídem, pero para revertir los cambios del script de su misma fila.
- Módulos
 - Lista de elementos añadidos, modificados y eliminados, y de observaciones si fuera necesario.
- Servicios externos expuestos
 - Lista de elementos añadidos, modificados y eliminados, y de observaciones si fuera necesario.
- Servicios consumidos de terceros
 - Lista de elementos añadidos, modificados y eliminados, y de observaciones si fuera necesario.
- Carpetas de intercambio
 - Lista de elementos añadidos, modificados y eliminados, y de observaciones si fuera necesario.

Ejemplo de tabla:

Tarea				Creación de ventana de gestión de usuarios			
Pertenece a		SPR-ACMEMAYHEM-0001	Número tarea		1		
Scripts							
	Scripts			Scripts de marcha atrás			
Independientes del entorno							
Dependientes del entorno							
Módulos							
	Nombre módulo		Observaciones				
Añadidos							
Modificados							
Eliminados							
Servicios externos expuestos							
	Nombre servicio		Observaciones				
Añadidos							
Modificados							
Eliminados							
Servicios consumidos de terceros							
	Nombre servicio		Observaciones				
Añadidos							

Modificados		
Eliminados		

Carpetas de intercambio		
	Nombre carpeta intercambio	Observaciones
Añadidos		
Modificados		
Eliminados		

La finalidad de estas es facilitar la actualización del diagrama de dependencias y de los detalles de los diferentes entornos del proyecto. Además, ayuda en las operaciones manuales que han de llevarse a cabo por los distintos grupos encargados de los despliegues.

4.2.1.3 *Tabla de resumen de pruebas por entorno*

Ruta documentación: /projects/[nombre corto proyecto]/testsbyenv

Si es la primera vez, se ha de crear una tabla para el proyecto con la lista de pruebas básicas (las que se usan para validar una subida) y exhaustivas (las realizadas para verificar totalmente las distintas funcionalidades) a realizar para cada entorno. Si no, revisar si hay que añadir, quitar o modificar alguna de ellas.

El código identificativo único de la tabla se formará con el prefijo TBE (Tests by Environment), un guion y el nombre corto del proyecto.

Ejemplo:

TBE-ACMEMAYHEM		Entorno		
Resumen pruebas básicas		DES	PRE	PRO
1	Login aplicación	Sí	Sí	Sí
2	Navegación entre diferentes secciones	Sí	Sí	Sí
3	Alta pedido	Sí	Sí	Sí
4	Consulta datos pedido	Sí	Sí	No
5	Creación informe pedidos	Sí	Sí	No
...
n
Resumen pruebas exhaustivas		DES	PRE	PRO

1	Prueba exhaustiva 1	Sí	Sí	Sí
2	Prueba exhaustiva 2	Sí	Sí	Sí
...
n

4.2.1.4 Descripción de las pruebas

Ruta documentación: /projects/[nombre corto proyecto]/testsdefinition

Los pasos detallados a realizar para las pruebas se guardarán en un documento aparte. Tanto el equipo de desarrollo como los probadores deben tener acceso a él. Además de la tabla de histórico de cambios y tener definido quién es el responsable de modificar el documento, se tendrá también una de datos comunes, que recogerá los parámetros que varían sus valores según el entorno. No hará falta crear un documento por entorno.

Ejemplo de tabla de datos comunes:

Datos comunes		
Campo/concepto	Entorno	Valor
Página de inicio	DES	http://servdes01:9080/mayhem
	PRE	http://servpre01:9080/mayhem
	PRO	http://mayhem.acme.com
Usuario estándar de pruebas y contraseñas	DES	user1/password1
	PRE	user1/password1
	PRO	support/

Las tablas que describen cada prueba incluirán los siguientes campos:

- Código identificativo único
 - Prefijo propuesto: TST (Test)
 - Formato: TST-[Nombre corto proyecto]-000
- Nombre corto
 - Debe ser suficientemente aclarativo como para saberse a cuál se refiere
 - Alfanumérico
- Aplicable a
 - Entornos a los que se puede aplicar esta prueba
- Descripción
 - Breve descripción de lo que se quiere probar
- Precondiciones
 - Condiciones o estado al que se debe haber llegado antes de comenzar la prueba
- Datos concretos de la prueba

- Fecha/hora de la prueba
 - En blanco en la plantilla. A rellenar cuando se realice la prueba.
- Usuario probador
 - En blanco en la plantilla. A rellenar con el usuario que realice la prueba.
- Pasos
 - Cada paso tendrá definido
 - un **número de paso**
 - una **descripción**
 - unos **datos de entrada**, que son los que se usarán para realizar la acción indicada en la descripción
 - un **resultado esperado**
 - columnas de resultado (**OK/KO**), que se irán marcando al realizarse las pruebas
- Observaciones sobre los errores encontrados
 - En el caso de encontrar algún error, es conveniente anotar en este campo información útil para su posterior revisión. Por ejemplo, trazas de error o capturas de pantalla.

Ejemplos:

TST-ACMEMAYHEM-000		Comprobación de servicios expuestos			
Tipo prueba		Básica	Aplicable a		<input checked="" type="checkbox"/> DES <input checked="" type="checkbox"/> PRE <input checked="" type="checkbox"/> PRO
Descripción		Verificar que están levantados los servicios web o REST de la aplicación.			
Precondiciones					
N/A					
Datos concretos de la prueba					
Fecha/hora de la prueba		Usuario probador			
Pasos					
Nº	Descripción	Datos entrada	Resultado esperado	OK	KO
1	Abrir SoapUI o Postman	N/A	N/A	<input type="checkbox"/>	<input type="checkbox"/>
2	Importar fichero de tests a realizar	Fichero de tests de servicios web o REST	N/A	<input type="checkbox"/>	<input type="checkbox"/>
3	Ejecutar pruebas	N/A	Todos los tests OK.	<input type="checkbox"/>	<input type="checkbox"/>
Resultado prueba				<input type="checkbox"/>	<input type="checkbox"/>

Observaciones sobre los errores encontrados

TST-ACMEMAYHEM-001 Login aplicación					
Tipo prueba		Básica	Aplicable a		<input checked="" type="checkbox"/> DES <input checked="" type="checkbox"/> PRE <input checked="" type="checkbox"/> PRO
Descripción		Probar que funciona correctamente el inicio de sesión de la aplicación.			
Precondiciones					
N/A					
Datos concretos de la prueba					
Fecha/hora de la prueba		Usuario probador			
Pasos					
Nº	Descripción	Datos entrada	Resultado esperado	OK	KO
1	Abrir navegador y acceder a la página de inicio	N/A	Se muestra la página de inicio	<input type="checkbox"/>	<input type="checkbox"/>
2	Introducir usuario de prueba y contraseña válidos	*Ver tabla de datos comunes	Página principal y menú con las diferentes secciones	<input type="checkbox"/>	<input type="checkbox"/>
Resultado prueba				<input type="checkbox"/>	<input type="checkbox"/>
Observaciones sobre los errores encontrados					

TST-ACMEMAYHEM-002 Navegación entre diferentes secciones					
Tipo prueba		Básica	Aplicable a		<input checked="" type="checkbox"/> DES <input checked="" type="checkbox"/> PRE <input checked="" type="checkbox"/> PRO
Descripción		Probar que se puede acceder a todas las secciones de la aplicación web.			
Precondiciones					
Haber accedido como usuario estándar de prueba y estar en la página principal. Ver TST-ACMEMAYHEM-001					
Datos concretos de la prueba					
Fecha/hora de la prueba		Usuario probador			
Pasos					
Nº	Descripción	Datos entrada	Resultado esperado	OK	KO

1	Acceder a la sección de administración	N/A	Se muestra la página de administración	<input type="checkbox"/>	<input type="checkbox"/>
2	Acceder a la sección de pedidos	N/A	Se muestra la página de pedidos	<input type="checkbox"/>	<input type="checkbox"/>
3	Acceder a la sección de inventario	N/A	Se muestra la página de inventario	<input type="checkbox"/>	<input type="checkbox"/>
4	Volver a la sección principal	N/A	Se muestra la página principal	<input type="checkbox"/>	<input type="checkbox"/>
Resultado prueba				<input type="checkbox"/>	<input type="checkbox"/>
Observaciones sobre los errores encontrados					

TST-ACMEMAYHEM-003		Alta pedido			
Tipo prueba	Básica	Aplicable a	<input checked="" type="checkbox"/> DES	<input checked="" type="checkbox"/> PRE	<input type="checkbox"/> PRO
Descripción	Probar que el servicio de alta de pedidos funciona correctamente.				
Precondiciones					
1) Haber accedido como usuario estándar de prueba y estar en la página principal. Ver TST-ACMEMAYHEM-001					
2) Disponer de herramientas que permitan invocar servicios REST (por ejemplo: Postman, SoapUI).					
Datos concretos de la prueba					
Fecha/hora de la prueba		Usuario probador			
Pasos					
Nº	Descripción	Datos entrada	Resultado esperado	OK	KO
1	Introducir ejemplo de alta de pedido en la herramienta de invocación de servicios REST	*Ver enlace documentación de SER-ACMEMAYHEM-001	N/A	<input type="checkbox"/>	<input type="checkbox"/>
2	Configurar la petición para que use la autenticación básica con el usuario de prueba estándar y agregar la contraseña, y ejecutar	*Ver tabla de datos comunes	Status: 201 (Created)	<input type="checkbox"/>	<input type="checkbox"/>
3	Acceder en la web a la sección de pedidos	N/A	Se muestra la página de pedidos	<input type="checkbox"/>	<input type="checkbox"/>
4	Realizar una búsqueda filtrando por el día actual, y ordenar por	Fecha: [Día de hoy]	En el listado debe aparecer el pedido que hemos dado de alta en primer lugar	<input type="checkbox"/>	<input type="checkbox"/>

	fecha descendente				
5	Hacer clic sobre el pedido y comprobar que los datos del mismo se corresponden con los de la llamada	N/A	Se muestra la pantalla del pedido y los datos coinciden con los usados en la llamada al servicio REST	<input type="checkbox"/>	<input type="checkbox"/>
				Resultado prueba	<input type="checkbox"/>
Observaciones sobre los errores encontrados					

Cada vez que se quieran realizar las pruebas básicas en un entorno de pruebas, se deberá crear un nuevo documento que deberá incluir una sección en la que se especifique:

- *Sprint* al que se refieren
- *Build* del código
- Entorno sobre el que se ejecutan
- Fecha de inicio de las pruebas

Las tablas a usar para las pruebas serán las mismas que las definidas en este mismo apartado. A medida que se realicen, se rellenarán los “datos concretos de la prueba”, las columnas de OK/KO (hasta el primer KO o el final), el resultado de la prueba, y escribir si procediera las observaciones sobre los errores encontrados.

El documento resultante se guardará en la ruta correspondiente a la documentación de pruebas:

/projects/[nombre corto proyecto]/sprints/[nº sprint]/tests

4.2.1.5 No conformidad

Ruta documentación: /projects/[nombre corto proyecto]/ncr

Según el punto 8.7.2 de la norma ISO 9001:2015 (6), la organización debe conservar la información documentada que:

- a) describa la no conformidad
- b) describa las acciones tomadas
- c) describa las concesiones obtenidas
- d) identifique la autoridad que decide la acción con respecto a la no conformidad

Para ello, se propone una plantilla con los siguientes campos:

- Código identificativo único
 - Prefijo propuesto: NCR (Non-Conformance Report)
 - Formato: NCR-[Nombre corto proyecto]-0000
- Sprint afectado
 - Código del *sprint* afectado
- Entorno
 - Marcar el entorno asociado a la no conformidad
- Fecha no conformidad
 - Fecha en la que sucede la no conformidad
- Hora no conformidad
 - Hora en la que sucede la no conformidad
- Fecha resolución
 - Fecha en la que se resuelve la no conformidad
- Hora no conformidad
 - Hora en la que se resuelve la no conformidad
- Detectado por
 - Indica quién ha comunicado la no conformidad
- Importancia
 - Nivel de importancia que da el propio cliente a la no conformidad
 - Los valores posibles son:
 - baja
 - media
 - alta
 - crítica
- Enlace a incidencia
 - Cuando se registre la incidencia asociada, se anotará aquí el enlace de la misma
- Descripción de la no conformidad
- Acciones tomadas
- Concesiones obtenidas
- Autoridad que decide la acción con respecto a la no conformidad

Plantilla propuesta:

NCR-ACMEMAYHEM-0001			
Sprint afectado		Entorno	<input type="checkbox"/> PRE <input type="checkbox"/> PRO
Fecha no conformidad		Hora no conformidad	
Fecha resolución		Hora resolución	
Detectado por			
Importancia	<input type="checkbox"/> Baja <input type="checkbox"/> Media <input type="checkbox"/> Alta <input type="checkbox"/> Crítica		
Enlace a incidencia			
Descripción de la no conformidad			
Acciones tomadas			
Concesiones obtenidas			
Autoridad que decide la acción con respecto a la no conformidad			

4.2.1.6 *Plan de marcha atrás*

Ruta documentación: /projects/[nombre corto proyecto]/rollbackplan

La sección de marcha atrás contendrá una tabla similar a la de pasos en la que aparezcan como filas las tareas que tienen una tarea de rollback específica. Compartirán el mismo ID para que así, si falla algo, sea fácilmente identificable la acción de reversión. Todas las anteriores a este punto también se tendrán que realizar

5 Subida a los distintos entornos

5.1 Sobre los despliegues y agrupación de tareas

La estrategia general debe consistir en subir lo antes posible al entorno de desarrollo los cambios asociados a tareas finalizadas. Como consecuencia, este entorno tendrá un número considerablemente mayor de despliegues que los que le siguen.

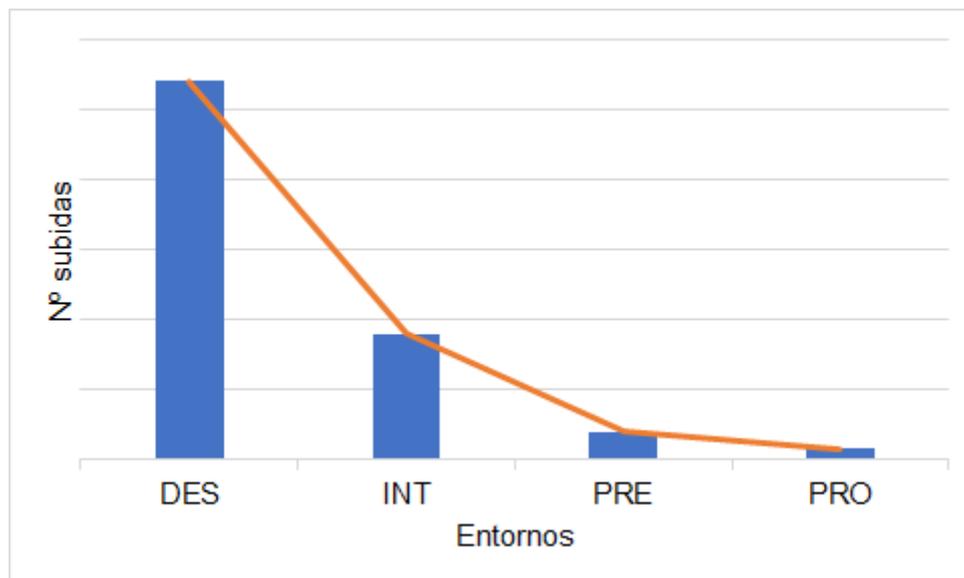


Ilustración 7: Frecuencia de despliegues por entorno (Excel). Basado en gráfico de www.if4it.com (7).

Nota: Los entornos más cercanos a PRO serán más estables, más fiables, y menos frecuentes las no disponibilidades por no estar continuamente desplegando.

Las tareas que se hayan validado en DES podrán ir promocionándose a entornos posteriores. Sin embargo, PRE se mantendrá como copia de PRO hasta que haya que preparar el entorno para que el cliente valide los nuevos cambios. Ese momento servirá como simulacro de subida a PRO. Además, tener la misma versión en PRE y PRO ayudará en la resolución de incidencias.

5.2 Tipos de subidas

Caso 1: Los desarrolladores tienen plenos poderes

Suele darse en los entornos de desarrollo. El equipo puede hacer y deshacer a su antojo. Al ser un entorno muy cambiante, los protocolos serán más flexibles, no siendo necesaria la confección de un plan de despliegue para cada vez. Además, no se necesitará de otros grupos, salvo que se quiera coordinar la subida con la de otro proyecto dependiente. En cualquier caso, se notificará la indisponibilidad temporal de la aplicación a los afectados.

Caso 2: Los desarrolladores dependen de terceros

En entornos como preproducción, el equipo de desarrollo no suele tener capacidad para modificar el modelo ni permisos de la base de datos, ni tampoco, los relativos a carpetas. Las peticiones han de hacerse a través de alguna web o aplicación tipo Jira, en donde se describirán las acciones necesarias. Se necesita adjuntar un plan de despliegue.

5.3 El subproceso de subida a DES/PRE/PRO

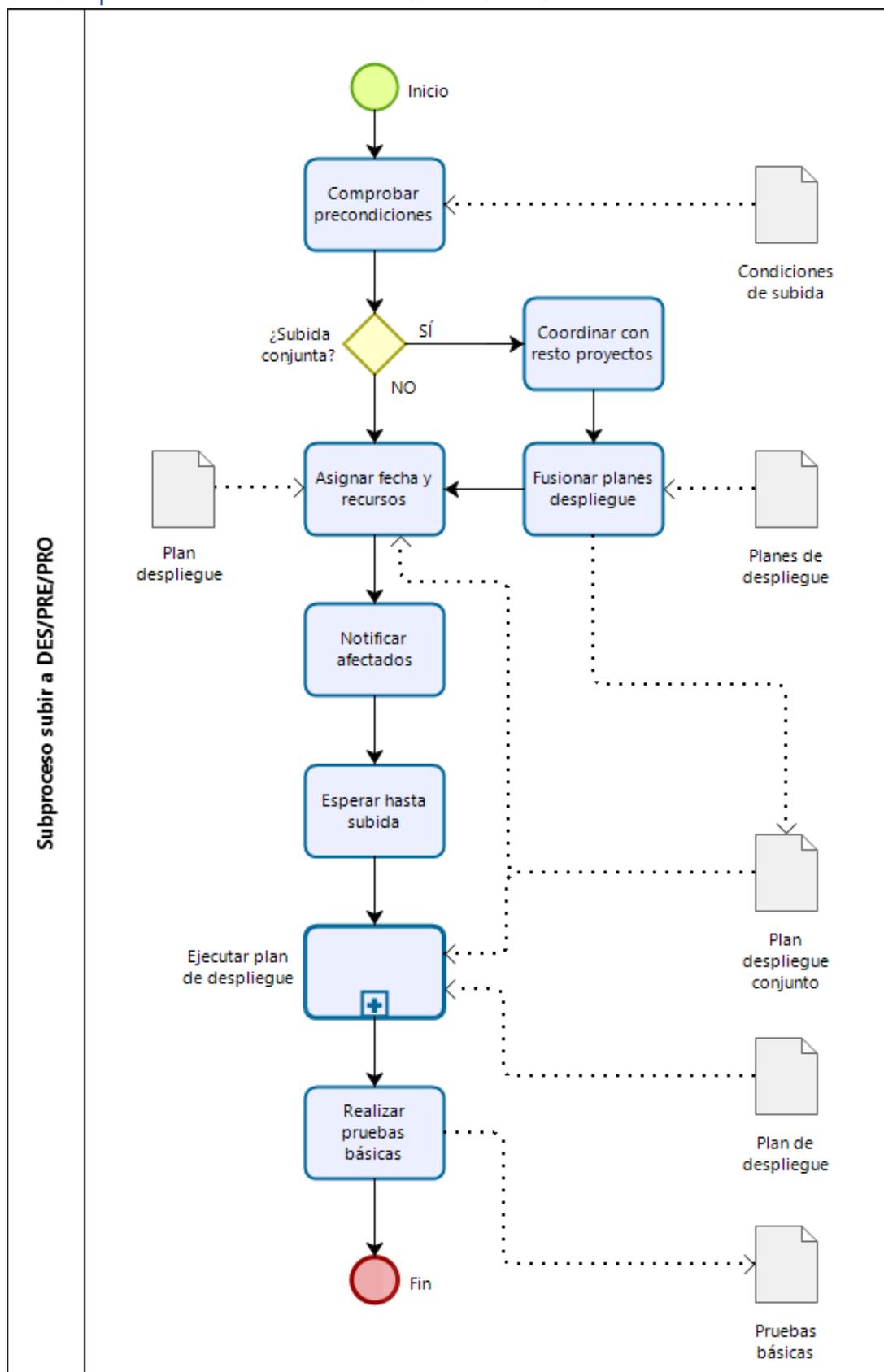


Ilustración 8: Diagrama de subproceso de subida a DES/PRE/PRO (Bizagi). Elaboración propia.

El subproceso de “subida a entorno” es válido para cualquiera de los entornos. Las diferencias principales:

	DES	PRE	PRO
Plan de despliegue necesario	No	Sí	Sí
Coordinación con otros proyectos si son afectados	Opcional	Sí	Sí
Flexibilidad de momento de despliegue	Alta	Baja	Ninguna

A continuación, los detalles de cada tarea del subproceso.

5.3.1 Comprobar precondiciones

5.3.1.1 Descripción

Se consulta si cumple las condiciones de subida para pasar de un entorno (ver tabla). Si no se cumplen, se esperará a las correcciones o se cancelará.

5.3.1.2 Responsables

- Equipo de desarrollo del proyecto

5.3.1.3 Documentos

Tipo	Documentos
Entrada	Condiciones de subida

5.3.1.4 Siguiete paso

¿Subida conjunta?	Siguiete paso
No	Asignar fecha y recursos
Sí	Coordinar con resto proyectos

5.3.2 Coordinar con resto de proyectos

5.3.2.1 Descripción

Si es una subida conjunta, se informará a los responsables de los otros proyectos implicados. Estos podrían decidir subir a la vez la parte que les corresponde (obligatorio si es PRO).

Nótese que toda subida conjunta es del *sprint* conjunto, pero en un *sprint* conjunto puede ser más conveniente subirlo por partes. Por ejemplo, cuando terceros necesitan que el proyecto que expone servicios lo tenga desplegado para probar su código. Puesto que no lo han probado, no tendrán codificada o probada su parte.

5.3.2.2 Responsables

- Equipo de desarrollo del proyecto

5.3.2.3 Documentos

N/A

5.3.2.4 Siguiete paso

- Fusionar planes despliegue

5.3.3 Fusionar planes despliegue

5.3.3.1 Descripción

Si es una subida conjunta, será necesario fusionar los planes de despliegue. ¿Qué ventajas tiene la fusión?

- Si falla algo del despliegue del proyecto que ha modificado un servicio, directamente se cancela la subida de los proyectos dependientes del mismo.
- Posibilidad de hacer más tareas en paralelo, aunque esto reduciría la ventaja del punto anterior.

5.3.3.2 Responsables

- Equipo de desarrollo del proyecto

5.3.3.3 Documentos

Tipo	Documentos
Entrada	Planes de despliegue
Salida	Plan despliegue conjunto

5.3.3.4 Siguiendo paso

- Asignar fecha y recursos

5.3.4 Asignar fecha y recursos

5.3.4.1 Descripción

Teniendo en cuenta:

- El tiempo estimado del despliegue incluido en el plan
- Franja de horario comprometido de las aplicaciones implicadas
- Los recursos disponibles
- Los datos aportados en la solicitud de subida
- Margen de tiempo que se necesitaría para una marcha atrás

...se decidirá la fecha y hora definitivas de la actuación.

A los técnicos no se les podrán asignar ninguna tarea prioritaria desde que realizan el primer paso hasta la finalización del despliegue. Las tareas de mantenimiento las podrán seguir realizando mientras estén a la espera de su siguiente actuación, pero en cuanto les toque el turno, han de dejar inmediatamente lo que estén haciendo.

5.3.4.2 Responsables

- Equipo de sistemas

5.3.4.3 Documentos

Tipo	Documentos
Entrada	Planes de despliegue

5.3.4.4 Siguiendo paso

- Notificar afectados

5.3.5 Notificar afectados

5.3.5.1 Descripción

Se enviará un aviso a las direcciones de correo electrónico de los proyectos involucrados, así como a los clientes en caso de que afecte a alguna de las aplicaciones dentro de su horario de uso habitual.

5.3.5.2 Responsables

- Equipo de desarrollo del proyecto

5.3.5.3 Documentos

N/A

5.3.5.4 Siguiente paso

- Esperar hasta subida

5.3.6 Esperar hasta subida

5.3.6.1 Descripción

Estado provisional hasta que llega la fecha y hora del despliegue.

5.3.6.2 Responsables

N/A

5.3.6.3 Documentos

N/A

5.3.6.4 Siguiente paso

- Ejecutar plan de despliegue

5.3.7 Ejecutar plan de despliegue

5.3.7.1 Descripción

Procedimiento de despliegue. Ver detalles en un apartado posterior.

5.3.7.2 Responsables

- Equipo de desarrollo del proyecto (si se responsabiliza de las pruebas básicas)
- Grupo de sistemas
- Grupo de base de datos
- Grupo de seguridad

5.3.7.3 Documentos

Tipo	Documentos
Entrada	Plan de despliegue o plan de despliegue conjunto

5.3.7.4 Siguiente paso

- Realizar pruebas básicas

5.3.8 Realizar pruebas básicas

5.3.8.1 Descripción

Verificación rápida de la aplicación en base al documento de pruebas rápidas.

5.3.8.2 Responsables

- Equipo de desarrollo del proyecto o grupo de sistemas

5.3.8.3 Documentos

Tipo	Documentos
Salida	Pruebas básicas

5.3.8.4 Siguiete paso

- [Fin]

5.4 El procedimiento “ejecutar plan de despliegue”

Dentro del subproceso de “subida a entorno”, este procedimiento es el más importante, por lo que detallaremos sus tareas.

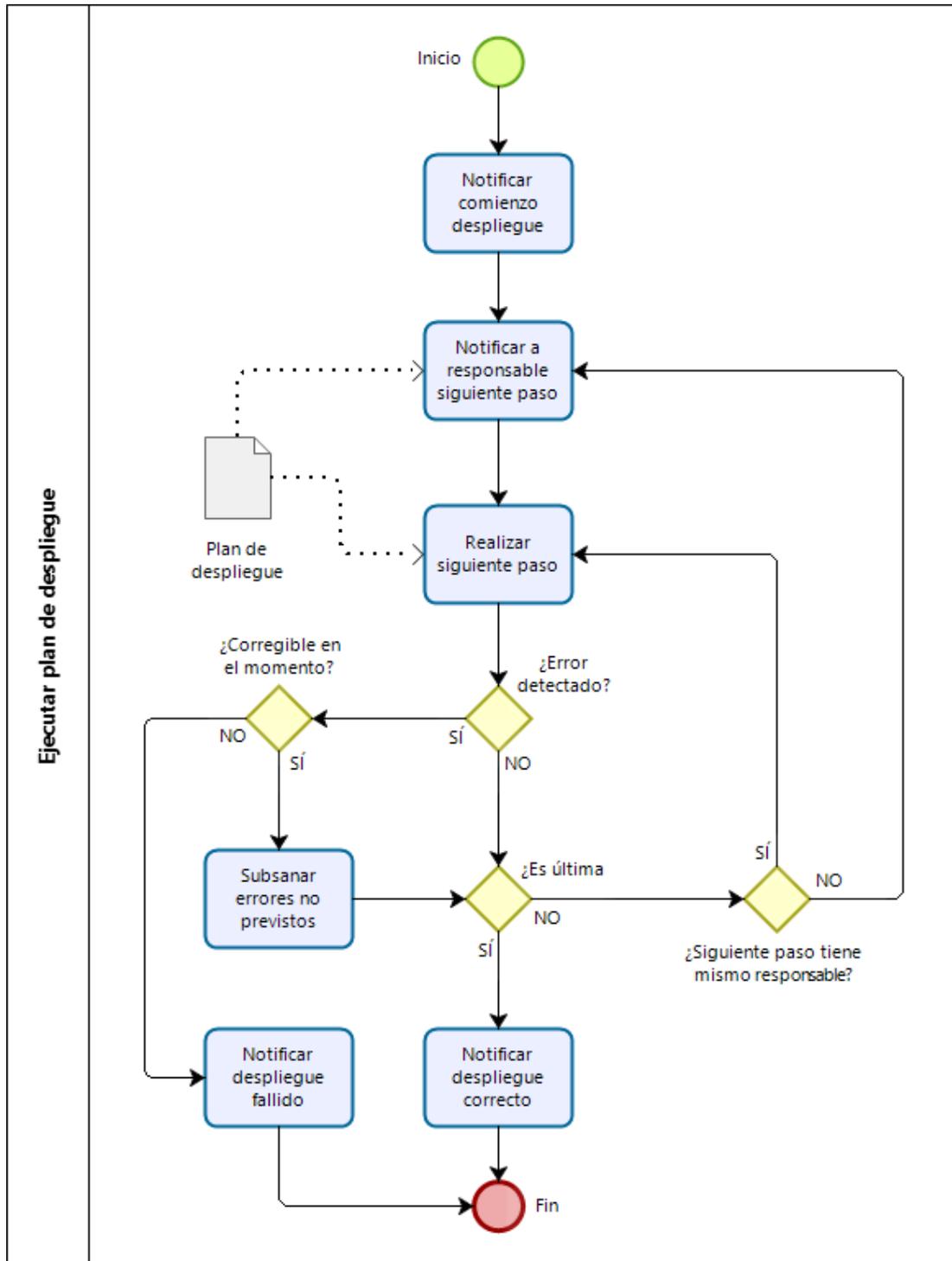


Ilustración 9: Diagrama de ejecutar plan de despliegue

Nota: Bizagi no da como válida la utilización de compuertas exclusivas (decisiones) con múltiples entradas y salidas a la vez, pero se ha usado así para simplificar el diagrama.

5.4.1 Notificar comienzo despliegue

5.4.1.1 Descripción

Se enviará un correo electrónico a todos los que participen en el despliegue indicando el comienzo del mismo.

5.4.1.2 Responsables

- Grupo de sistemas

5.4.1.3 Documentos

N/A

5.4.1.4 Siguiete paso

- Notificar a responsable de paso siguiente

5.4.2 Notificar a responsable de paso siguiente

5.4.2.1 Descripción

Siguiendo el plan de despliegue, tanto en el primer paso como para los siguientes en los que el encargado de la tarea sea diferente, se enviará una notificación (puede ser correo o cualquier otro sistema).

5.4.2.2 Responsables

Si es la primera, grupo de sistemas. Si no, quien haya completado el paso.

5.4.2.3 Documentos

Tipo	Documentos
Entrada	Plan de despliegue (conjunto o no)

5.4.2.4 Siguiete paso

- Realizar siguiente paso

5.4.3 Realizar siguiente paso

5.4.3.1 Descripción

Siguiendo un orden, se realizará la siguiente tarea (la primera, si aún no se hizo ninguna) siguiendo el plan de despliegue.

El número de puntos y causas por las que una subida puede no ser exitosa hace que sea imposible detallar cada uno de los escenarios, pero las estrategias a seguir suelen ser dos:

- Reintentar subida: si el código, configuración o scripts contenían algún error, podría intentar repetirse la subida desde el paso que se considere oportuno. Es posible que tengan que revertirse algunos cambios ya realizados.
- Marcha atrás: Antes de salir de este paso, deben ejecutarse en orden inverso todas las acciones descritas en el campo “reversión de cambios” del plan de despliegue.

La decisión de si reintentar o seguir ha de tomarla la persona responsable de nivel 2 del proyecto responsable del cambio.

5.4.3.2 Responsables

Depende de la tarea.

5.4.3.3 Documentos

Tipo	Documentos
Entrada	Plan de despliegue (conjunto o no)

5.4.3.4 Siguiete paso

¿Error detectado?	¿Corregible en el momento?	¿Es la última?	¿Siguiete paso tiene mismo responsable?	Siguiete paso
No	-	No	No	Notificar a responsable siguiete paso
No	-	No	Sí	Realizar siguiete paso
No	-	Sí	-	Notificar despliegue correcto
Sí	No	-	-	Notificar despliegue fallido
Sí	Sí	-	-	Subsanar errores no previstos

5.4.4 Notificar despliegue correcto

5.4.4.1 Descripción

Se enviará un correo electrónico a todos los que participan en el despliegue indicando la finalización del mismo.

5.4.4.2 Responsables

- Grupo de sistemas

5.4.4.3 Documentos

N/A

5.4.4.4 Siguiete paso

- [Fin]

5.4.5 Subsanar errores no previstos

5.4.5.1 Descripción

Se ha detectado un error subsanable. Realizar las correcciones oportunas y decidir si se puede continuar con el siguiete paso, o bien hay que retroceder hasta algún otro.

5.4.5.2 Responsables

Depende de la tarea.

5.4.5.3 Documentos

N/A

5.4.5.4 Siguiete paso

¿Es la última?	¿Siguiete paso tiene mismo responsable?	Siguiete paso
No	No	Notificar a responsable siguiete paso
No	Sí	Realizar siguiete paso
Sí	-	Notificar despliegue correcto

5.4.6 Notificar despliegue fallido

5.4.6.1 Descripción

Se envía un aviso a todos los participantes diciendo que el despliegue ha fallado.

5.4.6.2 Responsables

- Grupo de sistemas

5.4.6.3 Documentos

N/A

5.4.6.4 Siguiente paso

- [Fin]

5.5 El plan de despliegue

Ruta documentación (un proyecto): /projects/[nombre corto proyecto]/sprints/[nº sprint]/deploymentplan

Ruta documentación (más de uno): /projects/shared/[código sprint conjunto]/deploymentplan

Quizás, el documento más importante de todos, ya que es vital para los despliegues en entornos que necesiten de actuaciones de distintos grupos, sobre todo si es un *sprint* compartido con otros proyectos. Aunque el código modificado en cada iteración de desarrollo software no siempre afecta a los mismos módulos ni a los mismos servicios (si es que varía alguno), los pasos para el despliegue suelen ser parecidos. Siempre que sea posible, se tomará alguno del histórico como referencia.

Para la elaboración del plan, toman especial relevancia las tablas de las tareas del *sprint*, ya que de un vistazo se sabrá si se necesita o no ejecutar scripts de base de datos, los módulos que cambian, las operaciones necesarias sobre carpetas de intercambio...

Como norma general, el documento será lo más detallado posible dentro de la sencillez, y será preferible escribir una línea más a que una frase pueda resultar ambigua. Además, debe incluir:

- Códigos de los proyectos implicados
- Códigos de los *sprint* únicos implicados
- Código de *sprint* conjunto (si procede)
- Resumen de los grupos implicados en el despliegue
- Enlace a las pruebas básicas de los proyectos implicados
- Tabla de pasos

Tabla de pasos

Cada paso descrito en el documento deberá tener al menos estos campos:

- ID
 - Identificador único del paso.
- Grupo responsable
 - Grupo al que corresponde realizar la acción.
- Acciones
 - Conjunto de acciones a realizar en este paso suficientemente detalladas como para que un técnico no tenga dudas sobre cómo actuar.
- Comprobaciones

-
- Observaciones que pueden facilitar que la acción se ha realizado correctamente.
 - Si no es válida, valorar si son necesarias las acciones de “reversión de cambios”.
 - Reversión de cambios
 - Acciones a realizar en caso de tener que revertir los cambios de este paso.
 - Suele usarse siempre que haya una modificación en la base de datos, puesto que, si hay algún error, se debe dejar tal cual estaba antes.
 - ID siguiente
 - ID del siguiente paso que debe realizarse (si no ha habido errores). En algunos casos, pueden ser varios (si se optara por paralelismo), o “N/A”, si fuera el último.
 - Tiempo estimado
 - Duración estimada de la actuación.
 - Tiempo total
 - Suma de los tiempos estimados.

Plantilla de tabla de pasos:

ID	Grupo responsable	Acciones	Comprobaciones	Reversión de cambios	ID siguiente	Tiempo estimado
1					2	
2					3	
3					4	
...					...	
n					N/A	
						Tiempo total (min)

Nota: Para acortar los tiempos de despliegue, los pasos pueden procesarse con algún programa (Microsoft Project u otros similares) para, una vez hechas las asignaciones, conseguir el mayor paralelismo posible.

6 Cuadro de mando y mejora continua

El cuadro de mando (8) es una herramienta visual de gestión que permite medir la evolución de una lista de objetivos previamente definidos, facilitando así la toma de decisiones en caso detectarse desviaciones y/o comportamientos anómalos. Cada proyecto que siga esta guía debe tener implementado uno.

Desde el punto de vista del procedimiento de subidas, los aspectos a mejorar suelen estar asociados al tiempo empleado en estas. La dedicación de recursos a un despliegue en producción tiene un coste elevado, y le suele suponer al cliente la indisponibilidad del producto o servicio contratado en caso de que se requiera una parada. Por otra parte, los proyectos que se adaptan a la metodología de esta guía pueden tener un déficit en cuanto a documentación. Existirá un desfase que habrá que intentar disminuir y, por tanto, se supervisará.

Consideraciones y frecuencia de actualización

- Los datos del cuadro de mando y los diagramas asociados a cada uno de los objetivos se actualizarán lo antes posible después cada despliegue de un *sprint* en producción.
- Los *hotfix* no se incluirán en el estudio puesto que la carga de trabajo asociada suele ser mucho menor, y podría desvirtuar la serie.

6.1 Objetivos recomendados

Sin perjuicio de otros que se consideren oportunos según la naturaleza y peculiaridades de cada proyecto, se procede a describir los objetivos estándar, indicadores, representaciones gráficas que facilitan el seguimiento y detección de anomalías (9), así como una lista de posibles medidas correctivas en caso de presentarse alguna de estas.

6.1.1 Reducción del tiempo medio de despliegues

6.1.1.1 Definición

Tal y como se ha comentado con anterioridad, la reducción del tiempo medio de despliegues en producción será uno de los principales objetivos a conseguir, pero distinguiremos entre los casos: en los que la subida sólo es del proyecto, y en los que se implique también a otros. En esta última categoría cuentan tanto los *sprints* originados por el proyecto como los que no.

6.1.1.2 Indicador

Como indicador, tomaremos la media de minutos empleados para hacer los despliegues del año actual. Los minutos de referencia iniciales pueden ser los de la media del año anterior, o en su defecto, el estimado de la primera subida. El objetivo será reducir la cifra en un porcentaje que se considere razonable.

6.1.1.3 Representación

Para explotar la información, además del cuadro de mando se puede optar por gráficos de líneas con valores parciales. Los dos propuestos son:

1. Tiempo (en minutos) por cada despliegue
 - Permite ver la evolución temporal de cada *sprint* y posibles tendencias de bajada o subida de mejora de tiempos.
 - Incluye una línea de referencia que marca el tiempo objetivo

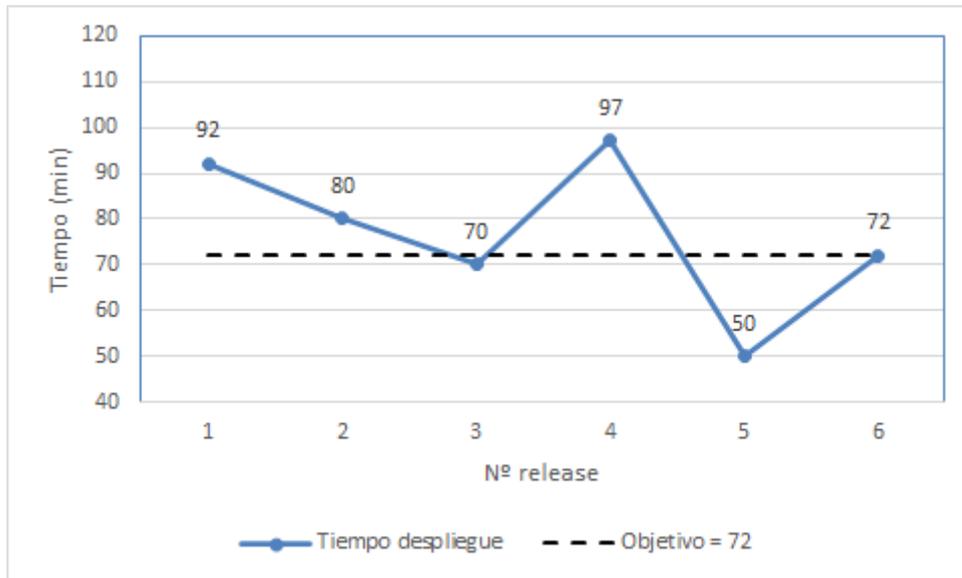


Ilustración 10: Gráfico de líneas de tiempo por despliegue. Elaboración propia.

2. Tiempo medio (en minutos) hasta *sprint* n

- Igual que el anterior, pero acumulando los tiempos del *sprint* n y sus anteriores y haciendo la media.
- Da menos información de cada *sprint*, pero deja más clara la cercanía al objetivo marcado, que es el tiempo medio (no el de cada *sprint*).

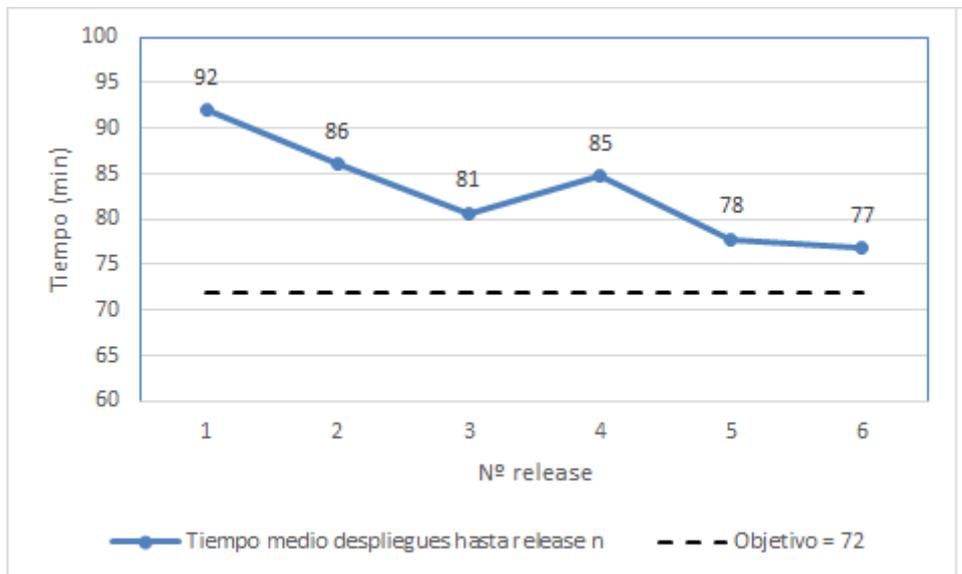


Ilustración 11: Gráfico de líneas de tiempo medio por despliegue hasta *sprint* n. Elaboración propia.

6.1.1.4 Interpretación y medidas correctivas

La tendencia lógica es que la línea tienda a acercarse a horizontal marcada como objetivo. Las subidas o los estancamientos por encima de esta son mala señal, y habría que realizarse estas preguntas:

Cuestiones a analizar	Medidas correctivas sugeridas
¿Se pueden automatizar pasos de los descritos en el documento de despliegue?	Rediseñar el documento de despliegue teniendo en cuenta estos puntos.
¿Se pueden hacer pasos en paralelo de los descritos en el documento de despliegue?	
¿Ha habido algún retraso por ambigüedad en algún paso de los descritos en el documento de despliegue?	Detallar más los pasos que corresponda.
¿Ha habido algún retraso por falta de capacitación de alguno de los técnicos asignados?	Asignar a otro técnico en futuros despliegues o mejorar su capacitación. También puede hacerse una doble asignación, de manera que tenga apoyo en caso de duda.

6.1.2 Reducción de casos en los que el despliegue se desvía de la estimación inicial más allá de un margen razonable

6.1.2.1 Definición

Además de reducir el tiempo de despliegue, no es menos importante el reducir las desviaciones con respecto a lo inicialmente estimado. Tardar mucho más de lo previsto puede afectar a otras tareas (además del coste extra). Tardar mucho menos, aunque es menos grave, puede significar tener desasignado al personal.

Se establecerán los porcentajes a partir de los cuales se considera significativa la desviación. Habrá una entrada para las positivas y otra para las negativas.

6.1.2.2 Indicador

El indicador es el número de veces que se supera el margen de error (por encima o por debajo, según corresponda).

6.1.2.3 Representación

Para explotar la información, además del cuadro de mando se puede optar por un diagrama de barras en el que se represente la desviación del tiempo de despliegue con respecto al estimado para cada *sprint*. Dos líneas horizontales marcarán el máximo y el mínimo admisibles.

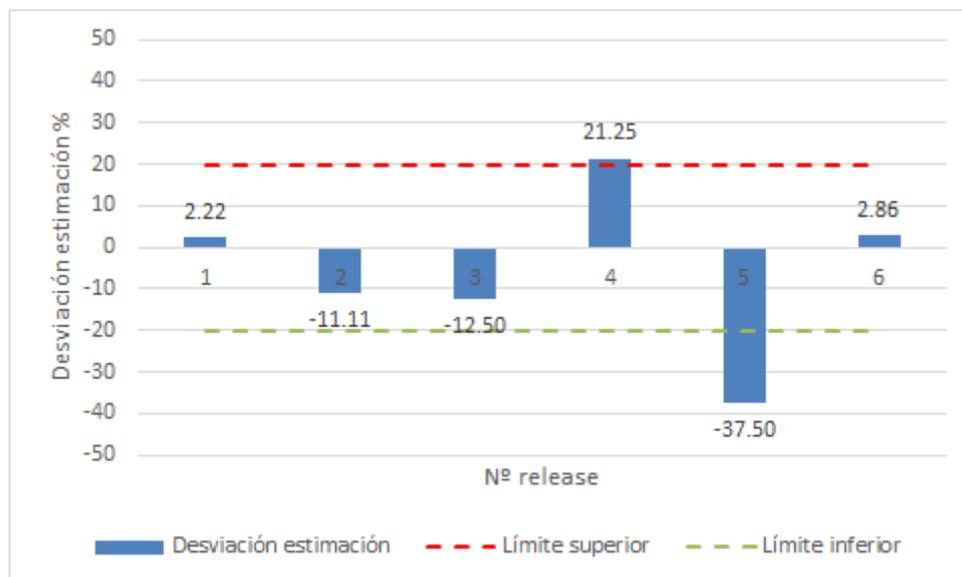


Ilustración 12: Diagrama de barras de desviación de estimación. Elaboración propia.

6.1.2.4 Interpretación y medidas correctivas

Las barras deberían estar siempre entre las líneas marcadas como límites. Cuando se sobrepase el máximo, hay que seguir esta tabla:

Cuestiones a analizar	Medidas correctivas sugeridas
¿Ha habido algún retraso por ambigüedad en algún paso de los descritos en el documento de despliegue?	Detallar más los pasos que corresponda.
¿Ha habido algún retraso por falta de capacitación de alguno de los técnicos asignados?	Asignar a otro técnico en futuros despliegues o mejorar su capacitación. También puede hacerse una doble asignación, de manera que tenga apoyo en caso de duda.
¿Se ha infraestimado alguna de las tareas?	Subir el tiempo estimado a tareas iguales o similares para futuras subidas.

Si, por el contrario, se sobrepasa el mínimo:

Cuestiones a analizar	Medidas correctivas sugeridas
¿Se ha sobreestimado alguna de las tareas?	Disminuir el tiempo estimado a tareas iguales o similares para futuras subidas.

6.1.3 Documentación actualizada de los servicios expuestos a terceros

6.1.3.1 Definición

La documentación de los proyectos es algo que suele dejarse de lado, pues es un trabajo tedioso. A pesar de lo que pueda parecer al leer el Manifiesto Ágil (10), esta metodología no condena la documentación, sino la innecesaria. La documentación de los servicios expuestos a terceros es imprescindible, tanto para el conocimiento de los miembros del proyecto como para los que los usan en otros.

6.1.3.2 Indicador

Tomaremos como indicador el porcentaje de servicios expuestos a terceros debidamente documentados y actualizados.

6.1.3.3 Representación

La representación ideal en este caso es el gráfico de líneas.

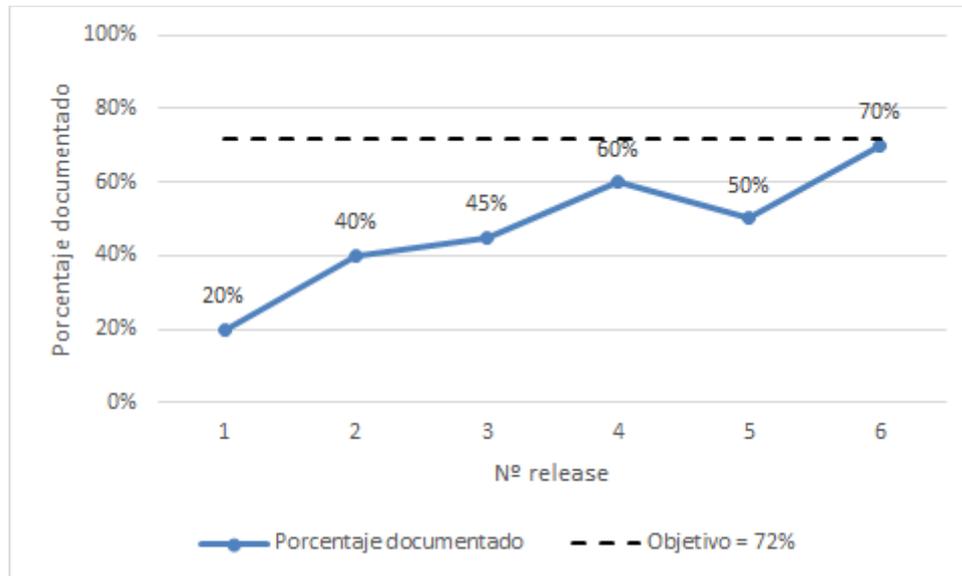


Ilustración 13: Porcentaje de documentación. Elaboración propia.

6.1.3.4 Interpretación y medidas correctivas

Todo lo que no sea aumentar el porcentaje se considera una mala señal. Aunque el objetivo acordado tenga una fecha límite, no se trata de dejarlo todo para el último momento. Es mejor que los incrementos sean constantes, o que incluso sean mayores al principio del periodo desde el que se monitoriza.

Cuestiones a analizar	Medidas correctivas sugeridas
¿Ha descendido el porcentaje? Se han creado nuevos servicios externos sin documentar.	Las tareas de cada sprint deben incluir la documentación si procede.
¿Sigue igual el porcentaje y está por debajo del objetivo?	Crear una tarea en el siguiente sprint que consista en documentar

6.2 Ejemplo de cuadro de mando

Cuadro de mando para proyecto ACMEMAYHEM (año 2020)

Responsable: Jefe de Proyecto (Mengano Santos)					Fecha última actualización: 01/06/2020					
Objetivos		Indicadores								
Descripción	Peso	Fórmula (unidades)	Procedimiento de obtención	Tipo	Peso en objetivo	Peso en CM	Umbral mínimo	Umbral máximo	Medida obtenida	Éxito
Reducción del tiempo medio de despliegues en producción en un 20%	40%	Minutos promedio por despliegue (un proyecto)	Consulta al departamento de sistemas, el cual tendrá registrados los tiempos estimados y reales de cada despliegue del proyecto.	Minimizador	60%	24%	60	48	55	41.67%
		Minutos promedio por despliegue (varios proyectos)		Minimizador	40%	16%	90	72	80	55.56%
Minimizar el número de veces en el que el tiempo de despliegue se desvía más de un 20% de lo estimado inicialmente	35%	Nº de despliegues en los que se produce una desviación positiva igual o superior al 20% de la estimación inicial	Consulta al departamento de sistemas, el cual tendrá registrados los tiempos estimados y reales de cada despliegue del proyecto.	Minimizador	75%	26%	2	0	1	50.00%
		Nº de despliegues en los que se produce una desviación negativa igual o superior al 20% de la estimación inicial		Minimizador	25%	9%	2	0	0	100.00%
Tener documentados y actualizados todos los servicios expuestos a terceros	25%	Porcentaje de servicios expuestos a terceros	Revisión de la documentación por parte del equipo del proyecto	Maximizador	100%	25%	80%	100%	90%	50.00%

Éxito global 53.26%

7 Conclusiones y valoración personal

Llegado este punto, vuelvo a leer tanto la introducción como el propósito y los objetivos de este TFM. ¿Realmente he conseguido lo que pretendía? ¿Con qué sensaciones me quedo? Al principio del todo temía que la temática no diera para rellenar más allá de unas treinta páginas, pero a medida que fui avanzando y profundizando, me di cuenta de que hay material para elaborar un libro de cientos de páginas. Crear una guía para el despliegue de servicios y productos no es tan sencillo.

La principal dificultad con la que me he encontrado es el tema de la gestión de la documentación. Tener un repositorio documental está bien, pero muchas de las tareas de un *sprint* generan documentos que pueden ir cambiando a lo largo del mismo y no siempre en los mismos puntos.

Por otro lado, el despliegue no se puede circunscribir exclusivamente a tener el código listo para su implantación; conlleva que se registren muchos elementos ajenos al proyecto que son esenciales. Otro problema es que el desarrollo no es secuencial. Mientras se está esperando un despliegue se sigue codificando o probando, lo que dificulta su encaje en un modelo tradicional de procesos.

El plan de despliegue también presenta dificultades por intentar ser genérico. Con un software a medida se podría conseguir mucha más agilidad entre pasos, e incluso conseguir que se hicieran varios pasos en paralelo. Imagino una pantalla que mostrara las instrucciones a seguir, qué hacer, un botón de siguiente paso que avisaría al encargado de la próxima tarea, o un botón de incidencia si ocurre un imprevisto, etc. En ese sentido, tiene mucho potencial.

En todo caso, y desde mi punto de vista, la guía resalta la importancia de tener procesos y procedimientos. Saber qué, quién y cómo ha de hacerse algo mejora la eficiencia y eficacia. También facilitan las mediciones: si se siguen siempre los mismos pasos, los valores de determinados parámetros pasarán a ser comparables. De esto último se deduce la gran importancia de no saltarse los protocolos. Basta con una vez que se incumpla para perjudicar enormemente el sistema de mejora continua que se tenga implementado. Los datos dejan de ser representativos, y las decisiones de aplicar o no, y en qué grado medidas correctivas tendrán más probabilidades de no ser las óptimas (y en ocasiones, negativas).

Resumiendo, este trabajo me ha permitido conocer mejor cada paso necesario hasta una puesta en producción. Los pequeños detalles, que pasan a veces desapercibidos, también son claves en la consecución de los objetivos. Lejos quizás de ver que esta guía sea una referencia para cualquier empresa, sí que me ha proporcionado nuevos conocimientos para estar más preparado en caso de que tenga que ejercer el rol de jefe de proyectos informáticos.

Glosario

Build

Proceso de conversión del código fuente a una forma independiente que pueda ser ejecutada en un ordenador. También puede referirse al resultado de dicho proceso.

Clustering

Agrupación de servidores conectados entre sí que operan como si fuera uno solo. Las principales ventajas que ofrece son la alta disponibilidad y la posibilidad de balancear la carga.

Herramienta de ticketing

Herramienta usada para el seguimiento de incidencias y peticiones (ejemplo: Jira).

Hotfix

Modificación en el software destinada a la corrección rápida de un problema.

No conformidad

Incumplimiento de un requisito. (11)

Out of the box (software)

Software que puede ser usado desde que se instala sin necesidad de una configuración o modificación.

Pila de producto (Scrum)

Lista ordenada que describe a grandes rasgos las funcionalidades y requisitos que el cliente desea para su producto.

Planning poker

Método de estimación de esfuerzo y duración de tareas basado en el uso de cartas con valores numéricos. Muy usado en proyectos Scrum.

Pruebas unitarias

Pruebas automatizadas de fragmentos de código destinadas a verificar su funcionamiento de manera aislada.

Script (base de datos)

Secuencia de comandos destinada a modificar la estructura, datos u otros aspectos de las bases de datos.

Scrum

Marco de trabajo de metodología ágil.

SID

Identificador de la instancia de base de datos.

Velocidad del equipo

En proyectos Scrum, cantidad de trabajo realizada por el equipo en un *sprint*.

Bibliografía

1. Martínez S. Mundo.erp - Entornos y metodologías de desarrollo en el software de gestión. [Online]; 2014 [consultado el 1 de junio de 2020. Disponible en: <https://www.mundoerp.com/blog/entornos-y-metodologias-de-desarrollo-software-gestion/>].
2. Martínez S. Mundo.erp - Entornos y metodologías en la implementación de un sistema de gestión. [Online]; 2015 [consultado el 1 de junio de 2020. Disponible en: <https://www.mundoerp.com/blog/entornos-metodologias-implementacion-sistema-gestion/>].
3. Disruptive Library Technology Jester - Traditional Development/Integration/Staging/Production Practice for Software Development. [Online]; 2006 [consultado el 1 de junio de 2020. Disponible en: <https://dltj.org/article/software-development-practice/>].
4. SL, Iubaris Info 4 Media. Scrum Manager - Guía de formación; 2019. pág. 16,26.
5. AENOR, editor. Sistemas de gestión de la calidad - Requisitos (ISO 9001:2015); 2015. pág. 10.
6. AENOR, editor. Sistemas de gestión de la calidad - Requisitos (ISO 9001:2015); 2016. pág. 30.
7. The International Foundation for Information Technology. [Online]; 2009 [consultado el 1 de junio de 2020. Disponible en: https://www.if4it.com/SYNTHESIZED/Frameworks/DEPLOYMENT/deployment_framework.html].
8. Módulo 4: Seguimiento y Mejora de un SGC. Curso Calidad Informática: Universidad de Alcalá; 2020. pág. 20-23.
9. Módulo 4: Seguimiento y Mejora de un SGC. Curso Calidad Informática: Universidad de Alcalá; 2020. pág. 19-20.
10. Autores varios. Manifiesto for Agile Software Development. [Online]; 2001 [consultado el 1 de junio de 2020. Disponible en: <http://agilemanifesto.org/>].
11. AENOR, editor. Sistemas de gestión de la calidad - Fundamentos y vocabulario (ISO 9000:2015); 2015. pág. 26.