
<Nombre de la empresa>

<Nombre del Proyecto>
<Nombre de la Consultora>
Plan de Desarrollo de Software

Versión 0.91

[Nota: La siguiente plantilla se ha desarrollado para su uso con Rational Unified Process. El texto que se encuentra entre corchetes y presentado en estilo itálicas azul se ha incluido para proporcionar una guía para el autor y se debería borrar antes de la entrega del documento.]

[Hay que sustituir el texto resaltado con marcador amarillo por su equivalente en el proyecto de desarrollo y eliminar el resaltado]

[La versión del documento se actualizará según la iteración y la fase del proyecto]

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

Historial de Revisiones

Fecha	Versión	Descripción	Autor
02/01/2002	0.9	Versión preliminar como propuesta de desarrollo.	Patricio Letelier Torres
31/05/2013	0.91	Correcciones al documento de propuesta de desarrollo	Aldo Valdez Alvarado

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

Tabla de Contenidos

1. Introducción.....	4
1.1 Propósito	4
1.2 Alcance	4
1.3 Resumen.....	4
2. Vista General del Proyecto	5
2.1 Propósito, Alcance y Objetivos.....	5
2.2 Suposiciones y Restricciones	5
2.3 Entregables del proyecto	5
2.4 Evolución del Plan de Desarrollo del Software.....	7
3. Organización del Proyecto	8
3.1 Participantes en el Proyecto.....	8
3.2 Interfaces Externas	8
3.3 Roles y Responsabilidades.....	8
4. Gestión del Proceso.....	9
4.1 Estimaciones del Proyecto.....	9
4.2 Plan del Proyecto	9
4.2.1 Plan de las Fases	9
4.2.2 Calendario del Proyecto.....	11
4.3 Seguimiento y Control del Proyecto	16
5. Referencias	16

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

Plan de Desarrollo del Software

1. Introducción

Este Plan de Desarrollo del Software es una versión preliminar preparada como parte del proyecto de prácticas de la asignatura de Análisis y Diseño de Sistemas de Información de la Carrera de Informática de la Universidad Mayor de San Andrés. Este documento provee una visión global del enfoque de desarrollo basado en el Rational Unified Process.

El proyecto ha sido ofertado por Patricio Orlando Letelier Torres basado en una metodología de Rational Unified Process en la que únicamente se procederá a cumplir con las tres primeras fases que marca la metodología, constando únicamente en la tercera fase de dos iteraciones. Es importante destacar esto puesto que utilizaremos la terminología RUP en este documento. Se incluirá el detalle para las fases de Inicio y Elaboración. Además Aldo Valdez Alvarado incluyo el detalle de las fases de Construcción y Transición para dar una visión global de todo proceso.

El enfoque de desarrollo propuesto constituye una configuración del proceso RUP de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los artefactos (entregables) que serán generados. Este documento es a su vez uno de los artefactos de RUP.

1.1 Propósito

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto. En él se describe el enfoque de desarrollo del software.

Los usuarios del Plan de Desarrollo del Software son:

- El jefe del proyecto lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento.
- Los miembros del equipo de desarrollo lo usan para entender lo qué deben hacer, cuándo deben hacerlo y qué otras actividades dependen de ello.

1.2 Alcance

[Una breve descripción del alcance de este Plan de Desarrollo Software no del sistema de software que se esta desarrollando]

1.3 Resumen

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

Vista General del Proyecto — proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto.

Organización del Proyecto — describe la estructura organizacional del equipo de desarrollo.

Gestión del Proceso — explica los costos y planificación estimada, define las fases e hitos del proyecto y describe cómo se realizará su seguimiento.

Planes y Guías de aplicación — proporciona una vista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas.

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

2. Vista General del Proyecto

2.1 Propósito, Alcance y Objetivos

La información que a continuación se incluye ha sido extraída de las diferentes reuniones que se han celebrado con el stakeholder de la empresa desde el inicio del proyecto.

2.1.1 Propósito

[Una breve descripción del propósito del sistema de software que se está desarrollando]

2.1.2 Alcance

[Una breve descripción del alcance del sistema de software que se está desarrollando]

2.1.3 Objetivos

[Una breve descripción del objetivo del sistema de software que se está desarrollando]

2.2 Suposiciones y Restricciones

[Describe todas aquellas suposiciones que se realizan sobre el proyecto y se declaran las restricciones impuestas tales como restricciones temporales, de hardware, de personal, etc.]

2.3 Entregables del proyecto

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que proponemos para este proyecto.

Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración.

1) Plan de Desarrollo del Software

Es el presente documento.

2) Modelo de Casos de Uso del Negocio

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas entre otras.), permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se representa con un Diagrama de Casos de Uso usando estereotipos específicos para este modelo.

3) Modelo de Objetos del Negocio

Es un modelo que describe la realización de cada caso de uso del negocio, estableciendo los actores internos, la información que en términos generales manipulan y los flujos de trabajo (workflows) asociados al caso de uso del negocio. Para la representación de este modelo se utilizan Diagramas de Colaboración (para mostrar actores externos, internos y las entidades (información) que manipulan, un Diagrama de Clases para mostrar gráficamente las entidades del sistema y sus relaciones, y Diagramas de Actividad para mostrar los flujos de trabajo.

4) Glosario

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología consensuada. .

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

5) Modelo de Casos de Uso

El modelo de Casos de Uso presenta las funciones del sistema y los actores que hacen uso de ellas. Se representa mediante Diagramas de Casos de Uso.

6) Visión

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.

7) Especificaciones de Casos de Uso

Para los casos de uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, post-condiciones, flujo de eventos, requisitos no-funcionales asociados. También, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación gráfica mediante un Diagrama de Actividad.

8) Especificaciones Adicionales

Este documento capturará todos los requisitos que no han sido incluidos como parte de los casos de uso y se refieren requisitos no-funcionales globales. Dichos requisitos incluyen: requisitos legales o normas, aplicación de estándares, requisitos de calidad del producto, tales como: confiabilidad, desempeño, etc., u otros requisitos de ambiente, tales como: sistema operativo, requisitos de compatibilidad, etc.

9) Prototipos de Interfaces de Usuario

Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la fase de Elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.

10) Modelo de Análisis y Diseño

Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al avance del proyecto, para ello se utilizan diagramas de clase de análisis (producto del los casos de uso del negocio), y diagramas de clase de diseño (producto de los casos de uso del sistema).

11) Modelo de Datos

Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se parte de un Diagrama de Clases (donde se utiliza el mapeo de UML a un Modelado de Datos, para conseguir la representación de tablas, claves, y otros), puede usarse diagramas entidad – relación (modelo lógico) y diagramas relacionales (modelo físico).

12) Modelo de Implementación

Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema. (Este modelo es sólo una versión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento), se usa el diagrama de componentes.

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

13) Modelo de Despliegue

Este modelo muestra el despliegue la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes, aquí se usa el diagrama de nodos.

14) Casos de Prueba

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

15) Solicitud de Cambio

Los cambios propuestos para los artefactos se formalizan mediante este documento. Mediante este documento se hace un seguimiento de los defectos detectados, solicitud de mejoras o cambios en los requisitos del producto. Así se provee un registro de decisiones de cambios, de su evaluación e impacto, y se asegura que éstos sean conocidos por el equipo de desarrollo. Los cambios se establecen respecto de la última baseline (el estado del conjunto de los artefactos en un momento determinado del proyecto) establecida. En nuestro caso al final de cada iteración se establecerá una baseline.

16) Plan de Iteración

Es un conjunto de actividades y tareas ordenadas temporalmente, con recursos asignados, dependencias entre ellas. Se realiza para cada iteración, y para todas las fases.

17) Evaluación de Iteración

Este documento incluye le evaluación de los resultados de cada iteración, el grado en el cual se han conseguido los objetivos de la iteración, las lecciones aprendidas y los cambios a ser realizados.

18) Lista de Riesgos

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

19) Manual de Instalación

Este documento incluye las instrucciones para realizar la instalación del producto.

20) Material de Apoyo al Usuario Final

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: Guías del Usuario, Guías de Operación, Guías de Mantenimiento y Sistema de Ayuda en Línea

21) Producto

Los ficheros del producto empaquetados y almacenadas en un CD con los mecanismos apropiados para facilitar su instalación. El producto, a partir de la primera iteración de la fase de Construcción es desarrollado incremental e iterativamente, obteniéndose una nueva release al final de cada iteración.

Los artefactos 19, 20 y 21 se generarán a partir de la fase de Construcción, con lo cual se han incluido aquí sólo para dar una visión global de todos los artefactos que se generarán en el proceso de desarrollo.

2.4 Evolución del Plan de Desarrollo del Software

El Plan de Desarrollo del Software se revisará semanalmente y se refinará antes del comienzo de cada iteración.

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

3. Organización del Proyecto

3.1 Participantes en el Proyecto

[Se debe reemplazar los participantes expuestos aquí por los roles presentes en el RUP]

Jefe de Proyecto.

[Aquí se declara el perfil del candidato a este puesto, así como su nombre y apellidos]

Analista de Sistemas.

[Aquí se declara el perfil del candidato a este puesto, así como su nombre y apellidos]

Analistas - Programadores.

[Aquí se declara el perfil de los candidatos a estos puestos, así como sus nombres y apellidos]

Ingeniero de Software.

[Aquí se declara el perfil del candidato a este puesto, así como su nombre y apellidos]

3.2 Interfaces Externas

[Breve descripción de las interfaces y funcionalidad que ofrecerá el producto]

3.3 Roles y Responsabilidades

A continuación se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP.

[Se debe reemplazar los participantes expuestos aquí por los roles presentes en el RUP]

Puesto	Responsabilidad
Jefe de Proyecto	El jefe de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el jefe de proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto.
Analista de Sistemas	Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos.
Programador	Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario
Ingeniero de Software	Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación. Elaborar modelos de implementación y despliegue.

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

4. Gestión del Proceso

4.1 Estimaciones del Proyecto

El presupuesto del proyecto y los recursos involucrados se adjuntan en un documento separado.

4.2 Plan del Proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

4.2.1 Plan de las Fases

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase (para las fases de Construcción y Transición es sólo una aproximación muy preliminar)

[Se debe llenar la siguiente tabla de acuerdo al número de iteraciones y la duración prevista en el documento del RUP que se vio en clases]

Fase	Nro. Iteraciones	Duración
Fase de Inicio		
Fase de Elaboración		
Fase de Construcción		
Fase de Transición		

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

Descripción	Hito
Fase de Inicio	En esta fase desarrollarán los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento del Plan de Desarrollo del Proyecto. La aceptación del cliente /usuario del artefacto Visión y el Plan de Desarrollo marcan el final de esta fase.
Fase de Elaboración	En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera release de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. En nuestro caso particular, por no incluirse las fases siguientes, la revisión y entrega de todos los artefactos hasta este punto de desarrollo también se incluye como hito. La primera iteración tendrá como objetivo la identificación

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

	y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis / Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar si es necesario la planificación para asegurar el cumplimiento de los objetivos. Ambas iteraciones tendrán una duración de una semana.
--	---

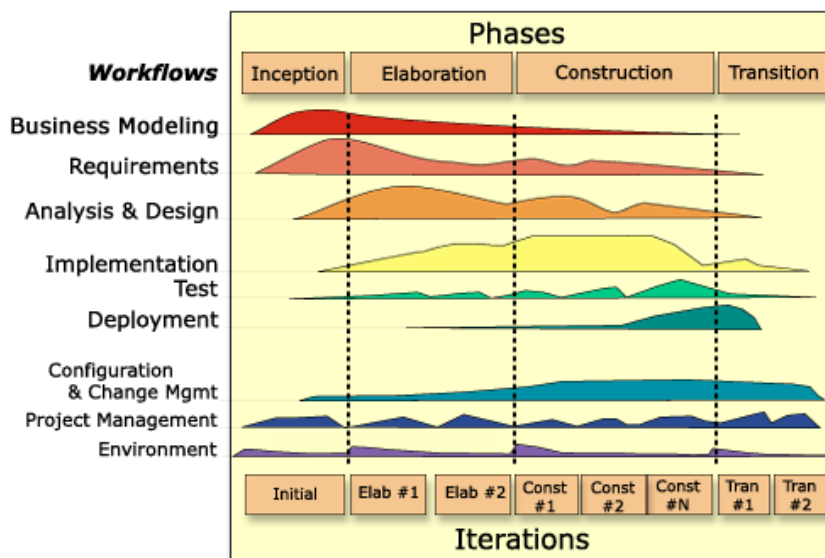
<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

Fase de Construcción	Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 2 iteraciones, cada una produciendo una release a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la versión de la release 2.0, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas beta.
Fase de Transición	En esta fase se prepararán dos releases para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto.

4.2.2 Calendario del Proyecto

A continuación se presenta un calendario de las principales tareas del proyecto incluyendo sólo las fases de Inicio y Elaboración. Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto. La siguiente figura ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina (workflow) en un momento determinado del desarrollo.

[Se debe reemplazar el siguiente gráfico en inglés por uno en español]



<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto en cuestión tiene un estado de completitud suficiente para someterse a revisión y aprobación, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

Disciplinas / Artefactos generados o modificados durante la Fase de Inicio	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio		
Requisitos		
Glosario		
Visión		
Modelo de Casos de Uso		siguiente fase
Especificación de Casos de Uso		siguiente fase
Especificaciones Adicionales		siguiente fase
Análisis/Diseño		
Modelo de Análisis/Diseño		siguiente fase
Modelo de Datos		siguiente fase
Implementación		
Prototipos de Interfaces de Usuario		siguiente fase
Modelo de Implementación		siguiente fase
Pruebas		
Casos de Pruebas Funcionales		siguiente fase
Despliegue		
Modelo de Despliegue		siguiente fase
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones		Siguiente fase
Ambiente	Durante todo el proyecto	

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

Disciplinas / Artefactos generados o modificados durante la Fase de Elaboración	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio		Aprobado
Requisitos		
Glosario		Aprobado
Visión		Aprobado
Modelo de Casos de Uso		Siguiente fase
Especificación de Casos de Uso		Siguiente fase
Especificaciones Adicionales		Siguiente fase
Análisis / Diseño		
Modelo de Análisis / Diseño		Revisar en cada iteración
Modelo de Datos		Revisar en cada iteración
Implementación		
Prototipos de Interfaces de Usuario		Revisar en cada iteración
Modelo de Implementación		Revisar en cada iteración
Pruebas		
Casos de Pruebas Funcionales		Revisar en cada iteración
Despliegue		
Modelo de Despliegue		Revisar en cada iteración
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 2.0 y planes de las Iteraciones		Revisar en cada iteración
Ambiente	Durante todo el proyecto	

Disciplinas / Artefactos generados o modificados durante la Fase de Construcción	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio		Aprobado
Requisitos		
Glosario		Aprobado

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

Visión		Aprobado
Modelo de Casos de Uso		Aprobado
Especificación de Casos de Uso		Aprobado
Especificaciones Adicionales		Aprobado
Análisis / Diseño		
Modelo de Análisis / Diseño		Siguiente fase
Modelo de Datos		Siguiente fase
Implementación		
Prototipos de Interfaces de Usuario		Siguiente fase
Modelo de Implementación		Siguiente fase
Pruebas		
Casos de Pruebas Funcionales		Siguiente fase
Despliegue		
Modelo de Despliegue		Revisar en cada iteración
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 2.0 y planes de las Iteraciones		Revisar en cada iteración
Ambiente	Durante todo el proyecto	

Disciplinas / Artefactos generados o modificados durante la Fase de Transición	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio		Aprobado
Requisitos		
Glosario		Aprobado
Visión		Aprobado
Modelo de Casos de Uso		Aprobado
Especificación de Casos de Uso		Aprobado
Especificaciones Adicionales		Aprobado
Análisis / Diseño		
Modelo de Análisis / Diseño		Aprobado
Modelo de Datos		Aprobado
Implementación		
Prototipos de Interfaces de Usuario		Aprobado
Modelo de Implementación		Aprobado
Pruebas		

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

Casos de Pruebas Funcionales		Aprobado
Despliegue		
Modelo de Despliegue		Aprobado a priori
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 2.0 y planes de las Iteraciones		Revisar en cada iteración
Ambiente	Durante todo el proyecto	

<Nombre del proyecto>	Versión: 0.91
Plan de Desarrollo de Software	Fecha: dd/mm/aaaa
Documento Plan de Desarrollo de Software	

4.3 Seguimiento y Control del Proyecto

Gestión de Requisitos

[Breve descripción de los requisitos que a los que se irá haciendo un seguimiento a lo largo de todo el proyecto]

Control de Plazos

[Figuran aquí los plazos de entrega de cada una de las fases planificadas]

Control de Calidad

[Descripción de los parámetros a tener en cuenta para llevar un control de calidad]

Gestión de Riesgos

[Definidos por el cliente]

Gestión de Configuración

[Resumen de los requisitos de configuración del producto generado en el proyecto]

5. Referencias

-
-
-