



Universidad Nacional de La Matanza
Departamento de Ingeniería e Investigaciones Tecnológicas

INGENIERÍA DE SOFTWARE

Trabajo Práctico Integrador

Grupo Systech

Alumnos

Apellido	Nombre	D.N.I.	E-mail
Benítez	Sabrina	29696633	sabrinaunlm@gmail.com
Coronado	Raúl	31192406	raul_jorge_coronado@yahoo.com.ar
Di Ludovico	Maximiliano	26122379	ivanahellsing@hotmail.com
Pirraglia	Walter E.	30603154	wpirraglia@gmail.com
Servis	Paola	30400372	paolaservis@yahoo.com.ar

FECHA DE PRESENTACIÓN: 23/07/2014

VERSIÓN: 1.0.0

Proyecto de desarrollo de software SISHO	4
Objetivos:	4
Funciones del sistema.....	4
Selección de un Modelo de ciclo de vida	4
Aplicación del método de selección de MCVS de Davis y Alexander	5
Tabla de selección de criterios y variables	5
Tabla según criterios según el proyecto	6
Clasificación para cada uno de los modelos	7
Modelo Convencional	7
Modelo Incremental	8
Modelo Evolutivo	9
Resultados.....	10
Ventajas.....	10
Desventajas	10
Representación gráfica del MCVS	10
Solución:	13
Mapa de actividades	13
Documentación adjunta a etapas para el desarrollo del producto software	16
Documentación en la etapa de Proceso de selección de un MVCS	16
Documentación en la etapa de Procesos de gestión del proyecto	16
Documentación en la etapa de PRE desarrollo del proyecto.....	16
Documentación en la etapa de Procesos de desarrollo.....	16
Documentación en la etapa de Post desarrollo del proyecto:.....	16
Documentación en la etapa de Procesos Integrales del Proyecto.....	16
Solución:.....	16
Estimación de tamaño.....	17
Tablas o Grupos de Entidad	19
Estimación de tamaño del software por medio de puntos de función.	19
Cálculo de PFSA	20
Estimación del Esfuerzo.....	20
Obtención de los PF ajustados.....	21
Calculo del esfuerzo, duración y cantidad de Personal	22
Modelo de métricas para la estimación de costos en proyectos de software.....	22
Factores de escala	22
Ajuste mediante drivers de costo.....	23
Multiplicadores de esfuerzo actualizados para el modelo de Diseño Anticipado	23
Nomenclatura para la interpretación de los resultados.....	29
Estimación del esfuerzo en forma manual:.....	30
Solución.....	32
Planificación del Proyecto	33
Distribución de fechas	38
Costo total	38
Gestion de riesgos	39
Gestión de Configuración.....	45
1. Identificación de la configuración.....	45
2. Definir líneas base	45
3. Establecer relaciones	45
4. Control de cambios de la configuración.....	45
5. Generación de informes de estado.....	45

6. Auditoría de la configuración	45
Modelos de calidad de producto	50
Métricas de Facilidad de Uso	50
Modelo de McCall.....	50
Relación de Factores con Criterios del Modelo McCall	52
Perfiles de cada puesto.....	58
Jefe de Proyecto.....	58
Analista Senior	59
Diseñador Senior	59
Programador Senior	60
Programador Semi Senior	60
Programador Junior.....	61
Tester	61
PROCESO DE PRODUCCIÓN.....	63
Análisis	63
Diseño	64
Codificación	64
Prueba	65
Mantenimiento	66
Proceso De Comercialización	67
Estudio de mercado.....	67
Análisis de clientes potenciales	68
Llamado.....	69
Visita.....	69
Seguimiento.....	70
Cierre de venta	71
Usabilidad del software	72
Solución.....	76

Proyecto de desarrollo de software SISHO

Sistema de Información de Administración Hotelera

Objetivos:

El motor que impulsa el desarrollo del sistema es la evidencia de una creciente complicación y dificultad en la gestión del manejo de habitaciones, reservas, ingresos y salida de huéspedes, generación de cargos, consultas de cuentas, consumos, generación de reportes e impresión de cuentas, que cada día aumenta debido al número de clientes del Hotel.

El futuro sistema no se encargará de la gestión de registro y reversión de pagos, quejas y reclamos y eliminación de cargos y facturas. Esas funciones las gestionará un software externo de acceso web que se contratará como un servicio externo, sin tener conexión con el sistema SISHO. Además, el sistema no interactuará con ningún otro sistema informático del Hotel.

Funciones del sistema

En términos generales, el sistema deberá proporcionar soporte a las siguientes tareas de gestión del Hotel:

1. Gestión de seguridad
2. Gestión de habitaciones
3. tarifas y planes
4. Gestión de Reservas
5. Gestión de Huéspedes
6. Generación de Consultas y Reportes
7. Gestión de Mantenimiento.

Selección de un Modelo de ciclo de vida

En función de la experiencia académica, laboral y el análisis del proyecto, el grupo tomó la decisión de seleccionar el Ciclo de Vida Iterativo, el sistema está muy bien definido, es grande y nos parece muy viable hacer entregas tempranas del producto software modularizadas que cumplan con los requerimientos del proyecto en forma iterativa e incremental hasta llegar al producto final, con la calidad inherente que se pretende del producto, porque de antemano podemos determinar que funcionalidades van a existir.

Por otra parte, este ciclo de vida es muy compatible con técnicas agilitas del mercado actual. Esta gran flexibilidad y adaptabilidad es otro factor importante es la elección de dicho Ciclo de vida.

Aplicación del método de selección de MCVS de Davis y Alexander.

Tabla de selección de criterios y variables

	Criterios	Vi1	Vi2	Vi3
C1	Experiencia de usuario	Principiante	Experimentado	Experto
C2	Expresión del usuario	Callado	Comunicativo	Expresivo
C3	Experiencia del equipo de desarrollo en la aplicación	Principiante	Experimentado	Experto
C4	Experiencia del equipo de desarrollo en la tecnología	Principiante	Experimentado	Experto
C5	Madurez de la aplicación	Nuevo	Standart	Estable
C6	Complejidad del problema	Simple	Difícil	Complejo
C7	Funcionalidad parcial	No deseable	Deseable	Urgente
C8	Frecuencia de los cambios	Raramente	Lento	Rápido
C9	Magnitud de los cambios	Mínimo	Moderado	Extremo
C10	Tamaño del producto	Pequeño	Medio	Grande
C11	Complejidad del producto	Simple	Difícil	Complejo
C12	Requerimientos no funcionales	Flexible	Moderado	Exacto
C13	Requerimientos de interfaz	Mínimo	Significativo	Crítico
C14	Etapa de asignación de fondos	Bajo-Alto	Estable	Alto-Bajo
C15	Disponibilidad de fondos	Escaso	Adecuado	Amplio
C16	Perfil del equipo	Bajo-Alto	Moderado	Alto-Bajo
C17	Disponibilidad del equipo	Escaso	Adecuado	Amplio
C18	Acceso a los usuarios	Nulo	Limitado	Libre
C19	Administración de la empresa compatible con el proyecto	Pautado	Flexible	Estricta
C20	Políticas de calidad de la empresa compatibles con el proyecto	Básico	Intermedio	Avanzado

Tabla según criterios según el proyecto

	Criterios	Vi1	Vi2	Vi3
C1	Experiencia de usuario	1	0	0
C2	Expresión del usuario	0	1	0
C3	Experiencia del equipo de desarrollo en la aplicación	1	0	0
C4	Experiencia del equipo de desarrollo en la tecnología	0	1	0
C5	Madurez de la aplicación	0	1	0
C6	Complejidad del problema	0	1	0
C7	Funcionalidad parcial	0	1	0
C8	Frecuencia de los cambios	0	1	0
C9	Magnitud de los cambios	0	0	1
C10	Tamaño del producto	0	0	1
C11	Complejidad del producto	0	1	0
C12	Requerimientos no funcionales	0	1	0
C13	Requerimientos de interfaz	0	0	1
C14	Etapas de asignación de fondos	-	-	-
C15	Disponibilidad de fondos	-	-	-
C16	Perfil del equipo	0	1	0
C17	Disponibilidad del equipo	0	0	1
C18	Acceso a los usuarios	0	1	0
C19	Administración de la empresa compatible con el proyecto	1	0	0
C20	Políticas de calidad de la empresa compatibles con el proyecto	0	1	0

Clasificación para cada uno de los modelos

Modelo Convencional

Criterio	Matriz del proyecto			Matriz de Selección Convencional			Cálculos			Suma
	Si1	Si2	Si3	Vi1	Vi2	Vi3	Si1*Vi1	Si2*Vi2	Si3*Vi3	
C1	1	0	0	0	0	1	0	0	0	0
C2	0	1	0	0	0	1	0	0	0	0
C3	1	0	0	0	0	1	0	0	0	0
C4	0	1	0	1	1	1	0	1	0	1
C5	0	1	0	0	0	1	0	0	0	0
C6	0	1	0	1	0	0	0	0	0	0
C7	0	1	0	1	0	0	0	0	0	0
C8	0	1	0	1	0	0	0	0	0	0
C9	0	0	1	1	0	0	0	0	0	0
C10	0	0	1	1	0	0	0	0	0	0
C11	0	1	0	1	0	0	0	0	0	0
C12	0	1	0	1	1	1	0	1	0	1
C13	0	0	1	1	0	0	0	0	0	0
C14	-	-	-	1	0	0				
C15	-	-	-	0	0	1				
C16	0	1	0	1	0	0	0	0	0	0
C17	0	0	1	0	0	1	0	0	1	1
C18	0	1	0	0	1	1	0	1	0	1
C19	1	0	0	1	1	1	1	0	0	1
C20	0	1	0	1	1	1	0	1	0	1
Total										6

Modelo Incremental

Criterio	Matriz del proyecto			Matriz de Selección Iterativo			Cálculos			Suma
	Si1	Si2	Si3	Vi1	Vi2	Vi3	Si1*Vi1	Si2*Vi2	Si3*Vi3	
C1	1	0	0	0	1	1	0	0	0	0
C2	0	1	0	0	1	1	0	1	0	1
C3	1	0	0	0	1	1	0	0	0	0
C4	0	1	0	0	1	1	0	1	0	1
C5	0	1	0	0	1	1	0	1	0	1
C6	0	1	0	1	1	0	0	1	0	1
C7	0	1	0	0	1	0	0	1	0	1
C8	0	1	0	1	1	0	0	1	0	1
C9	0	0	1	1	1	0	0	0	0	0
C10	0	0	1	1	1	1	0	0	1	1
C11	0	1	0	1	1	1	0	1	0	1
C12	0	1	0	1	1	0	0	1	0	1
C13	0	0	1	1	1	0	0	0	0	0
C14	-	-	-	0	1	0				
C15	-	-	-	0	1	1				
C16	0	1	0	0	1	0	0	1	0	1
C17	0	0	1	0	1	1	0	0	1	1
C18	0	1	0	0	1	1	0	1	0	1
C19	1	0	0	1	1	0	1	0	0	1
C20	0	1	0	0	1	1	0	1	0	1
Total										14

Modelo Evolutivo

Criterio	Matriz del proyecto			Matriz de Selección Evolutivo			Cálculos			Suma
	Si1	Si2	Si3	Vi1	Vi2	Vi3	Si1*Vi1	Si2*Vi2	Si3*Vi3	
c1	1	0	0	1	1	1	1	0	0	1
c2	0	1	0	1	1	1	0	1	0	1
c3	1	0	0	1	1	1	1	0	0	1
c4	0	1	0	0	1	1	0	1	0	1
c5	0	1	0	1	1	1	0	1	0	1
c6	0	1	0	1	1	1	0	1	0	1
c7	0	1	0	0	1	1	0	1	0	1
c8	0	1	0	1	1	1	0	1	0	1
c9	0	0	1	1	1	1	0	0	1	1
c10	0	0	1	1	1	1	0	0	1	1
c11	0	1	0	1	1	0	0	1	0	1
c12	0	1	0	1	0	0	0	0	0	0
c13	0	0	1	1	1	1	0	0	1	1
c14	-	-	-	0	1	0				
c15	-	-	-	1	1	1				
c16	0	1	0	0	1	0	0	1	0	1
c17	0	0	1	1	1	1	0	0	1	1
c18	0	1	0	0	0	1	0	0	0	0
c19	1	0	0	1	1	0	1	0	0	1
c20	0	1	0	0	1	1	0	1	0	1
Total										16

Luego de haber aplicado el criterio para la selección del Modelo de Ciclo de Vida de Alexander y Davis, comparando las tres matrices obtuvimos como resultado una cifra de clasificación de 14 para el Modelo Iterativo y una cifra de clasificación de 16 para el Modelo Evolutivo. No tenemos en cuenta la cifra de clasificación de 6 que corresponde al Modelo Convencional porque no aplica a nuestro Proyecto.

Si bien la diferencia entre el Modelo Iterativo y Evolutivo es mínima, de acuerdo a la experiencia del Líder del Proyecto y a la de nuestro equipo coincidimos en que el modelo indicado a aplicar es el Iterativo. La principal razón es porque está perfectamente definido el alcance, por lo tanto se puede organizar el desarrollo de las distintas funcionalidades en módulos (Iteraciones), que irán evolucionando a través de incrementos. Nuestro cliente irá obteniendo entregas parciales, con lo que podrá plantear modificaciones o mejoras a lo largo del Proyecto. Otro beneficio de este Modelo es que vamos a poder estimar y presupuestar todo el Proyecto.

Resultados

Modelo	Cifra de Clasificación
Convencional	6
Iterativo	14
Evolutivo	16

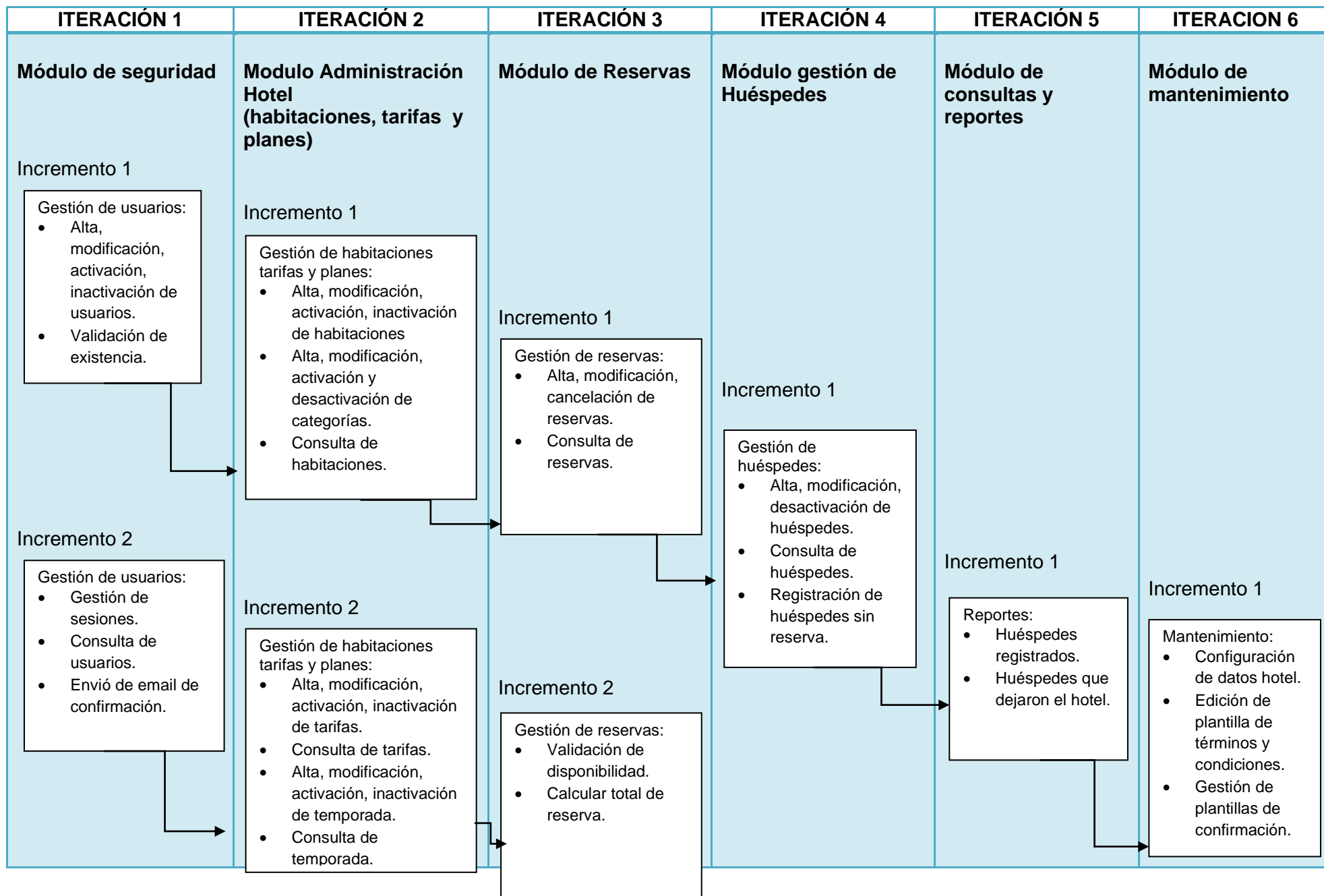
Ventajas

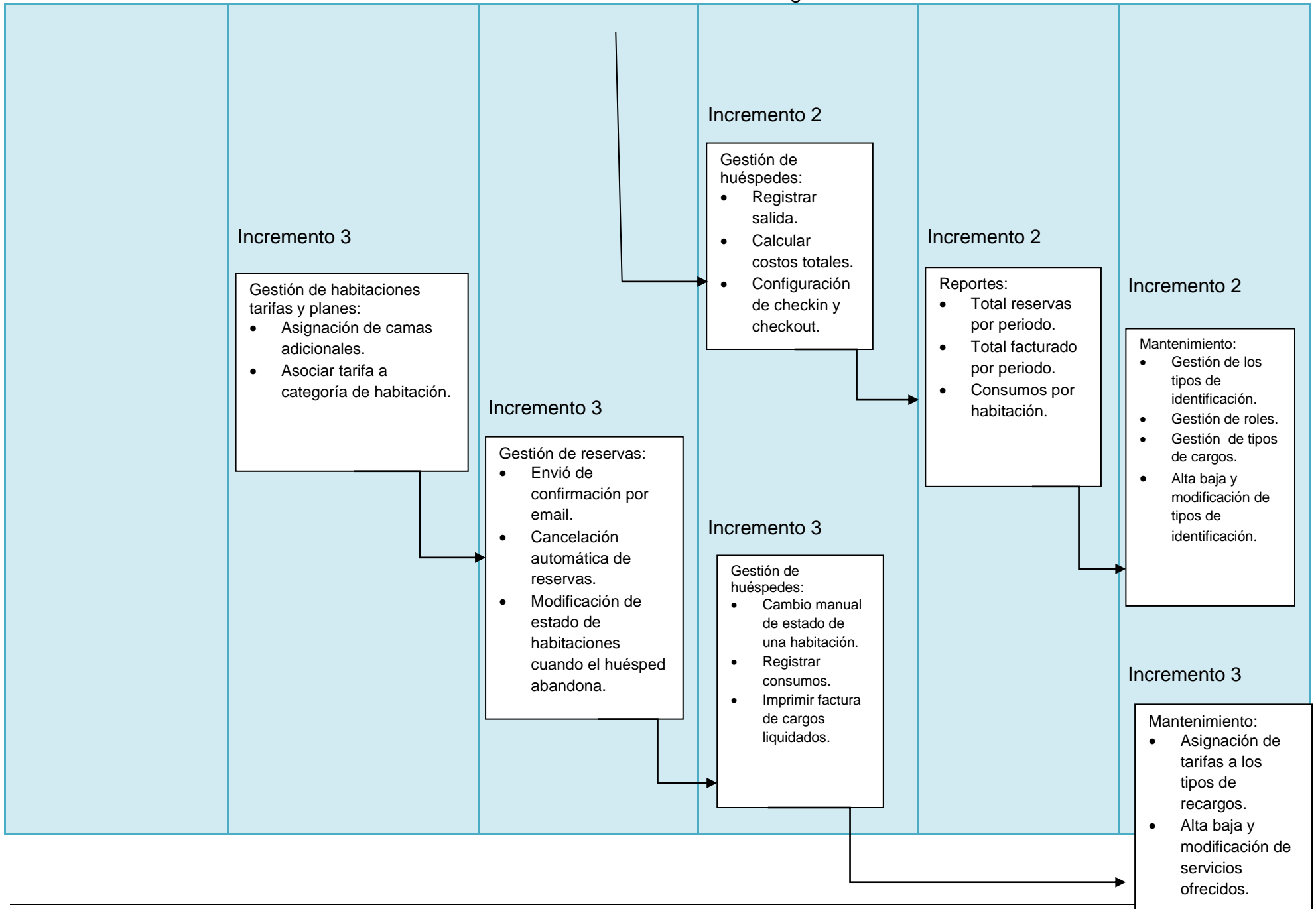
- Se reduce el tiempo de desarrollo inicial, ya que se implementa la funcionalidad parcial.
- Provee un impacto ventajoso frente al cliente, que es la entrega temprana de partes operativas del software.
- Resulta más sencillo acomodar cambios al acotar el tamaño de los incrementos.
- Los productos desarrollados con este modelo tienen una menor probabilidad de fallar.
- Se obtiene un aprendizaje en cada iteración que es aplicado en el desarrollo del producto y aumenta las experiencias para próximos proyectos

Desventajas

- El modelo incremental no es recomendable para casos de sistemas de tiempo real, de alto nivel de seguridad, de procesamiento distribuido y/o de alto índice de riesgos.
- Requiere de mucha planeación, tanto administrativa como técnica.
- Requiere de metas claras para conocer el estado del proyecto.
- Requiere de un cliente involucrado durante todo el curso del proyecto.

Representación gráfica del MCVS





Solución:

Para facilitar una metodología común entre el cliente y la empresa de software, los modelos de ciclo de vida se han actualizado para reflejar las etapas de desarrollo involucradas y la documentación requerida, de manera que cada etapa se valide antes de continuar con la siguiente etapa. El ciclo de vida permite que los errores se detecten lo antes posible y a los desarrolladores que se concentren en la calidad del producto. En el Modelo Iterativo se asume que los requisitos están claros, el alcance está bien definido desde el inicio.

Mapa de actividades

En un proyecto de desarrollo software debe seleccionarse el modelo de ciclo de vida que se utilizará y que guiará todo el proceso, para luego indicar que actividades específicas del proceso software se realizarán en cada una de las etapas. Esto implica definir un mapa de actividades que guiará todo el desarrollo del proyecto.

El Mapa de Actividades es una tabla de dos entradas, en la cual, una entrada es el proceso software con sus actividades y la otra entrada es el ciclo de vida seleccionado con sus etapas definidas. En el mapa de actividades deben marcarse con una cruz las actividades que se llevarán a cabo en cada una de las etapas del ciclo de vida.

Existen actividades que es necesario realizar una única vez, por ej. la selección de un modelo de ciclo de vida y existen otras actividades que se realizan en cada una de las etapas por ej. Gestión del Proyecto, documentación, etc.

ACTIVIDADES DE LOS PROCESOS		Incremento I	Incremento II	Incremento III	Incremento IV	Incremento V	Incremento VI
Proceso de Selección de un MCVS	Proceso de Selección de un MCVS						
	- Identificar los posibles MCVS	1					
	- Seleccionar un modelo para el proyecto.	2					
Procesos de Gestión del Proyecto	Proceso de Iniciación, Planificación y Estimación del Proyecto						
	- Establecer la matriz de actividades para el MCVS.	3					
	- Asignar los recursos del proyecto.	4	4	4	4	4	4
	- Definir el entorno del proyecto.	5	5	5	5	5	5
	- Planificar la gestión del proyecto.	5					
	Proceso de Seguimiento y Control del Proyecto						
	- Analizar riesgos.	6	6	6	6	6	6
	- Realizar la planificación de contingencias.	7					
	- Gestionar el proyecto.	8	8	8	8	8	8
	- Implementar el sistema de informes de problemas.	7					
	- Archivar registros.	7	7	7	7	7	7

	Proceso de Gestión de Calidad del Software					
	- Planificar la garantía de calidad del software.	9				
	- Desarrollar métricas de calidad.	9				
	- Gestionar la calidad del software.	9	9	9	9	9
	- Identificar necesidades de mejora de la calidad.		9	9	9	9
Proceso s de Pre - Desarroll o	Proceso de Exploración de Conceptos					
	- Identificar las ideas o necesidades.	10				
	- Formular las soluciones potenciales.	10				
	- Dirigir los estudios de viabilidad.	10				
	- Refinar y Finalizar la idea o necesidad.	10				
	Proceso de Asignación del Sistema					
	- Analizar las funciones del sistema.	11	11			
	- Desarrollar la arquitectura del sistema.	11	11			
	- Descomponer los requisitos del sistema.	11				
Proceso s de Desarroll o	Proceso de Análisis de Requisitos					
	- Definir y Desarrollar los requisitos del software.	12	12	12		
	- Definir los requisitos de interfaz.	13				
	- Priorizar e Integrar los requisitos del software.	12				
	Proceso de Diseño					
	- Realizar el diseño preliminar.	12				
	- Analizar el flujo de información.	13	13	13	13	13
	- Diseñar la base de datos (si se aplica).	13	13	13	13	13
	- Diseñar las interfaces.	13	13	13	13	13
	- Realizar el diseño detallado.		13	13	13	13
	Proceso de Implementación e Integración					
	- Crear los datos de prueba.	14	14	14	14	14
	- Crear el código fuente.		15	15	15	15
	- Generar el código objeto.		15	15	15	15
	- Crear la documentación de operación.		16	16	16	16
	- Planificar la integración.		16	16	16	16
	- Realizar la integración.		16	16	16	16
	Proceso de Instalación y Aceptación					
	- Planificar la instalación.	17	17	17	17	17
	- Distribuir el software.		17	17	17	17
	- Instalar el software.		17	17	17	17

o	- Aceptar el software en el entorno de operación.		17	17	17	17	17
	- Realizar las actualizaciones.		17	17	17	17	17
	Proceso de Operación y Soporte						
	- Operar el sistema.				18	18	18
	- Proveer de asistencia técnica y consultas.		18	18	18	18	18
	Proceso de Mantenimiento						
	- Realizar el mantenimiento correctivo.		19	19			
	- Reaplicar el ciclo de vida del software.				19	19	19
Procesos Integrales del Proyecto	Proceso de Verificación y Validación						
	- Planificar la verificación y validación.	20	20	20	20	20	20
	- Ejecutar las tareas de verificación y validación.	20	20	20	20	20	20
	- Recoger y Analizar los datos de las métricas.		20	20			
	- Planificar las pruebas.	20	20	20			
	- Desarrollar las especificaciones de las pruebas.	20	20	20			
	- Ejecutar las pruebas.		20	20			
	Proceso de Configuración						
	- Planificar la gestión de configuración.	20					
	- Realizar la identificación de la configuración.		20	20			
	- Realizar el control de la configuración.	20	20	20	20	20	20
	- Realizar la información del estado de la configuración.		20	20			
	Proceso de Documentación						
	- Planificar la documentación.	20					
	- Implementar la documentación.		20	20			
	- Producir y Distribuir la documentación.		20	20			
	Proceso de Formación						
	- Planificar el programa de formación.	20					
	- Desarrollar los materiales de formación.		20				
	- Validar el programa de formación.		20	20			
	- Implementar el programa de formación.			20			
	Tabla. Matriz de Actividades - Estandar IEEE 1074						

Documentación adjunta a etapas para el desarrollo del producto software

Documentación en la etapa de Proceso de selección de un MVCS:

- Documentación de especificación del modelo de MCVS (1)
- Acta de soporte para la elección del MCVS.(2)

Documentación en la etapa de Procesos de gestión del proyecto:

- Formulario de matriz de actividades.(3)
- Documentación de gestión de Planificación de recursos y estimación de proyecto.(4)
- Acta de gestión de metodologías y herramientas (5)
- Protocolo de análisis de riesgos potenciales(6)
- Reporte para planes de contingencia.(7)
- Expediente de Gestión de proyecto.(8)
- Expediente de métricas potenciales y aseguramiento de la calidad.(9)

Documentación en la etapa de PRE desarrollo del proyecto:

- Archivo de viabilidad y factibilidad del proyecto(10)
- Documento de estructuras y arquitectura(11)

Documentación en la etapa de Procesos de desarrollo:

- Legajo de análisis de requisitos.(12)
- Documentación de Diseño de interfaz(13)
- Expediente de pruebas.(14)
- Documentación de código.(15)
- Documentación para la integración y reportes.(16)

Documentación en la etapa de Post desarrollo del proyecto:

- Manual de distribución, instalación y actualización (17)
- Manual de usuario (18).
- Manual de mantenimiento.(19)

Documentación en la etapa de Procesos Integrales del Proyecto:

- Documentación sobre soporte del sistema (20).

Solución:

Como puede verse, el mapa marca qué actividades del proceso software deberán realizarse

en cada una de las etapas del ciclo de vida. Existen actividades que es necesario realizarlas una única vez (por ejemplo, la selección de un modelo de ciclo de vida). Sin embargo, existen otras actividades que se realizan en cada una de las etapas (por ejemplo,

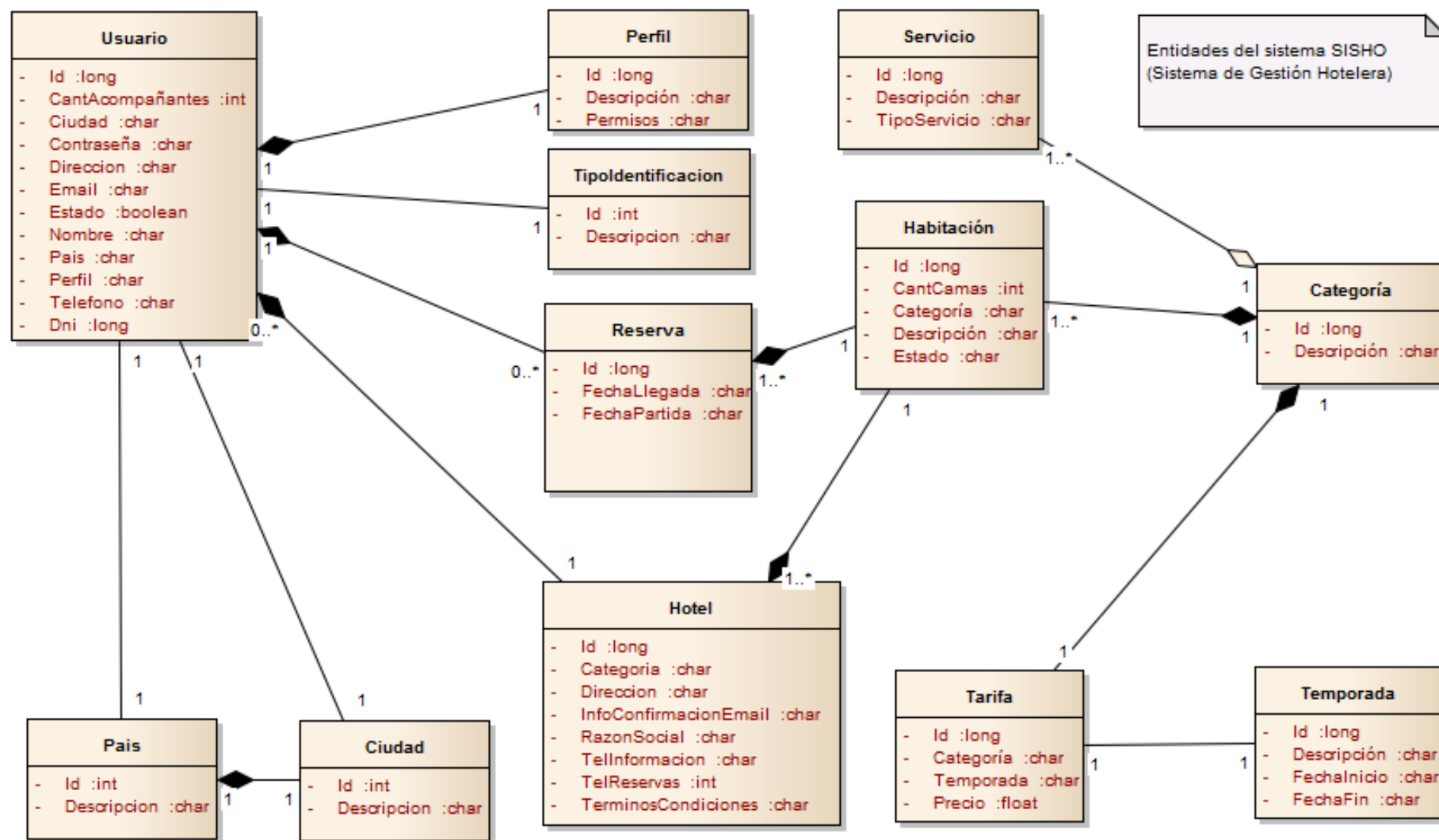
gestionar el proyecto). Obviamente, dependiendo del ciclo de vida elegido el mapa de actividades variará.

Por lo tanto, el mapa de actividades es el primer paso para conseguir una organización del proyecto que lleve a su gestión. A partir del mapa, puede pasarse a una estimación del tiempo y costo de cada una de las actividades, y por lo tanto, del proyecto global; a una asignación de recursos para cada actividad, etc.

Estimación de tamaño

A partir del Documento de Especificación de Requisitos debe realizarse la estimación del tamaño del producto aplicando la técnica de Puntos de Función. La estimación debe calcularse sobre la totalidad de los requisitos del documento y puede separarse por módulos del producto, por fases del proceso de desarrollo y realizar la estimación del producto en general. El objetivo es que se obtenga el cálculo estimado de puntos de función sin ajustar.

class Class Model



Tablas o Grupos de Entidad

Tabla o Grupo de Entidad	
1	Usuario - Perfil - Pais - Ciudad - Tipoidentificación
2	Habitación
3	Categoría - Tarifa
4	Temporada
5	Servicio
6	Hotel
7	Reserva

Estimación de tamaño del software por medio de puntos de función.

Tabla o Grupo de Entidad		Entrada			Salida			Consulta		
		ALTA	MEDIA	BAJA	ALTA	MEDIA	BAJA	ALTA	MEDIA	BAJA
1	Usuario - Perfil - Pais - Ciudad - Tipoidentificación			1			1			1
2	Habitación			1			1			1
3	Categoría - Tarifa		1				1		1	
4	Temporada			1			1			1
5	Servicio			1			1			1
6	Hotel			1			1			1
7	Reserva	1			1			1		

Cálculo de PFSA

Componente	Complejidad	Cantidad	x	Peso	Total
Entrada					
	Alta	1	x	6	6
	Media	1	x	4	4
	Baja	5	x	3	15
Salida					
	Alta	1	x	7	7
	Media	0	x	5	0
	Baja	6	x	4	24
Consulta					0
	Alta	1	x	6	6
	Media	1	x	4	4
	Baja	5	x	3	15
Fichero Lógico Interno					
	Alta	7	x	15	105
	Media	0	x	10	0
	Baja	0	x	7	0
Fichero Lógico Externo					
	Alta	0	x	10	0
	Media	0	x	7	0
	Baja	0	x	5	0
				PFSA =	186

Inicialmente realizamos un análisis aproximado de cada una de las funcionalidades del sistema, para poder establecer tanto los atributos como las entidades que están implicadas en cada caso.

Esto paso fue necesario para llevar adelante la aplicación del método de estimación de tamaño por Puntos de Función.

Finalmente obtuvimos como valor de PFSA: 186

Estimación del Esfuerzo

A partir del Documento de Especificación de Requisitos debe realizarse la estimación y del cálculo de estimación del tamaño de producto realizada con puntos de función, se deberá estimar el esfuerzo requerido para el proyecto en cantidad de personas por mes (meses/hombres).

Debe obtener como resultado el cálculo estimado de esfuerzo aplicando el modelo de estimación COCOMOII, utilizando los factores y variables del modelo. El valor a calcular debe representar el total de horas/hombre estimadas para todo el proyecto.

Obtención de los PF ajustados

Factor de Ajuste	Puntaje
Comunicación de Datos	2
Proceso Distribuido	0
Objetivos de Rendimiento	2
Configuración de Explotación Compartida	2
Tasa de transacciones	1
Entrada de Datos en Línea	5
Eficiencia con el Usuario Final	3
Actualizaciones en Línea	5
Lógica de Proceso Interno Compleja	3
Reusabilidad del Código	4
Conversión e Instalación contempladas	3
Facilidad de Operación	4
Instalaciones Múltiples	3
Facilidad de Cambios	3
TOTAL	40

SIGNIFICADO DEL VALOR DE CADA F.C.

Valor	Significado del valor
0	Sin influencia, factor no presente
1	Influencia insignificante, muy baja
2	Influencia moderada o baja
3	Influencia media, normal
4	Influencia alta, significativa
5	Influencia muy alta, esencial

PFSA	186
FC	40

$$PFA = PFSA * (0,65 + (0.01 * FC))$$

PFA: puntos de función ajustados.

PFSA: puntos de función sin ajustar.

FC: factor de ajuste.

$$PFA: 186 * (0,65 + (0,01 * 40)) = 195,3$$

Lenguaje	Correlación Codigo Fuente por PF	Tabla de Conversión de: Correlación Código Fuente a PF
Assembler	320	62496
C	128	24998.4
Fortran	105	20506.5
Pascal	91	17772.3
Basic	64	12499.2
Sql	13	2538.9
Html	14	2734.2
Java	46	8983.8
Delphi	118	23045.4
C++	53	10350.9
Cobol	107	20897.1

Si para el proyecto elegimos el lenguaje de programación java entonces tenemos:

$$\text{LOC}=46*195,3$$

$$\text{LOC}=8983,8$$

$$\text{KLOC}=8983,8/1000$$

$$\text{KLOC}= 9$$

Calculo del esfuerzo, duración y cantidad de Personal

Tipo de Proyecto	a	b	c	d
Orgánico	2.4	1.05	2.5	0.38
Medio	3.0	1.12	2.5	0.35
Embebido	3.6	1.20	2.5	0.32

Se utiliza para obtener una primera aproximación rápida del esfuerzo y hace uso de la tabla de constantes para calcular distintos aspectos de costos:

$$E = a(\text{KLOC})^b$$

$$D = c(E)^d$$

$$P = E/D$$

Esfuerzo (E) = $3,0*(9)^{1,12} = 35,14$ meses/hombre
Duracion (D) = $2,5*(35,14)^{0,35} = 8,68$ meses
Personal (P) = $35,14/8,68 = 4,04$ personas

Si asumimos que es un proyecto de **mediano plazo**.

El esfuerzo se calcula para una efectiva planeación y control, seleccionar los componentes adecuados y limitar las áreas del problema.

Modelo de métricas para la estimación de costos en proyectos de software

Herramienta Cocomo II (modelo constructivo de costo):

Es una herramienta para la estimación del esfuerzo, la duración y los costos involucrados en el desarrollo de un producto software. Estudia los tiempos y costos que se generan en el desarrollo del software. Esta basado sobre los puntos de función (que medirán el tamaño de un software) y que se convierten en líneas de código.

Factores de escala

Se refiere a un ajuste fino en:

Precedentedness: grado de los datos históricos de la organización.

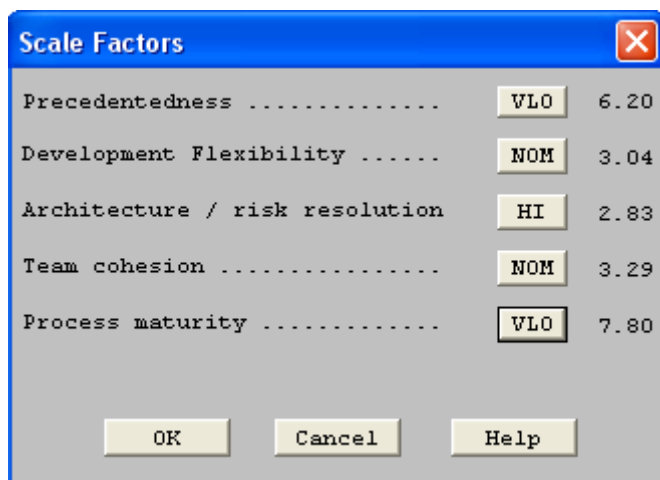
Development Flexibility: grado de flexibilidad del proyecto.

Architecture /risk reolution: grado de cambios que puede tener el proyecto.

Team cohesión: experiencia que el equipo tiene desarrollado estos proyectos.

Process maturity: claridad en el desarrollo de lo procesos.

Estos valores inciden sobre el valor de los costos.



The dialog box titled "Scale Factors" contains the following data:

Factor	Scale	Value
Precedentedness	VLO	6.20
Development Flexibility	NOM	3.04
Architecture / risk resolution	HI	2.83
Team cohesion	NOM	3.29
Process maturity	VLO	7.80

Buttons: OK, Cancel, Help

Ajuste mediante drivers de costo

	Extra Bajo	Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
RCPX	0.73	0.81	0.98	1.00	1.30	1.74	2.38
RUSE	--	--	0.95	1.00	1.07	1.15	1.24
PDIF	--	--	0.87	1.00	1.29	1.81	2.61
PERS	2.12	1.62	1.26	1.00	0.83	0.63	0.50
PREX	1.59	1.33	1.12	1.00	0.87	0.71	0.62
FCIL	1.43	1.30	1.10	1.00	0.87	0.73	0.62
SCED	--	1.43	1.14	1.00	1.00	1.00	--

Multiplicadores de esfuerzo actualizados para el modelo de Diseño Anticipado

Aplicando los valores de la tabla obtendremos:

Driver de costos para la iteración I:

EAF - Mod seguridad

base + incr % = rating

	RCPX	RUSE	PDIF	PERS	PREX	FCIL	USR1	USR2
base	LO	LO	NOM	HI	HI	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%	0%	0%	0%

EAF is also affected by Schedule

EAF: 0.57

OK Cancel Help

SLOC Input Dialog - Mod seguridad

Sizing Method

☐ SLOC

☒ Function Points

☐ Adaptation and Reuse

Breakage
% of code thrown away due to requirements evolution and volatility

REVL 0.00

Module Size in Function Points

Language JAVA Change Multiplier 40

Function Type	# of Function Points			SubTotal
	Low	Average	High	
Internal Logical Files	0	0	1	15
External Interface Files	0	0	0	0
External Inputs	1	0	0	3
External Outputs	1	0	0	4
External Inquiries	1	0	0	3
Total Unadjusted Function Points				25
Equivalent Total in SLOC				1000

OK Cancel Help

Driver de costos para la iteración II:

EAF - Mod Administracion

base + incr % = rating

	RCPX	RUSE	PDIF	PERS	PREX	FCIL	USR1	USR2
base	LO	LO	NOM	XLO	VLO	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%	0%	0%	0%

EAF is also affected by Schedule

EAF: 2.22

OK Cancel Help

SLOC Input Dialog - Mod Administracion

Sizing Method

☐ SLOC

☒ Function Points

☐ Adaptation and Reuse

Breakage

% of code thrown away due to requirements evolution and volatility

REVL 0.00

Module Size in Function Points

Language ▼ JAVA Change Multiplier 40

Function Type	# of Function Points			SubTotal
	Low	Average	High	
Internal Logical Files	0	0	2	30
External Interface Files	0	0	0	0
External Inputs	2	1	0	10
External Outputs	3	0	0	12
External Inquiries	0	0	0	0
Total Unadjusted Function Points				52
Equivalent Total in SLOC				2080

OK Cancel Help

Driver de costos para la iteración III:

EAF - Mod Reservas

base + incr % = rating

	RCPX	RUSE	PDIF	PERS	PREX	FCIL	USR1	USR2
base	LO	LO	NOM	LO	LO	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%	0%	0%	0%

EAF is also affected by Schedule

EAF: 1.11

OK Cancel Help

SLOC Input Dialog - Mod Reservas

Sizing Method

☐ SLOC

☒ Function Points

☐ Adaptation and Reuse

Breakage

% of code thrown away due to requirements evolution and volatility

REVL 0.00

Module Size in Function Points

Language ▼ JAVA Change Multiplier 40

Function Type	# of Function Points			SubTotal
	Low	Average	High	
Internal Logical Files	0	0	1	15
External Interface Files	0	0	0	0
External Inputs	0	0	1	6
External Outputs	0	0	1	7
External Inquiries	0	0	1	6
Total Unadjusted Function Points				34
Equivalent Total in SLOC				1360

OK Cancel Help

Driver de costos para la iteración IV:

EAF - Mod Gestion de huesp

base + incr % = rating

	RCPX	RUSE	PDIF	PERS	PREX	FCIL	USR1	USR2
base	LO	LO	NOM	LO	LO	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%	0%	0%	0%

EAF is also affected by Schedule

EAF: 1.11

OK Cancel Help

SLOC Input Dialog - Mod Gestion de huesp

Sizing Method

☐ SLOC

☒ Function Points

☐ Adaptation and Reuse

Breakage

% of code thrown away due to requirements evolution and volatility

REVL 0.00

Module Size in Function Points

Language JAVA Change Multiplier 40

Function Type	# of Function Points			SubTotal
	Low	Average	High	
Internal Logical Files	0	0	1	15
External Interface Files	0	0	0	0
External Inputs	0	0	0	0
External Outputs	0	0	0	0
External Inquiries	2	1	0	10
Total Unadjusted Function Points				25
Equivalent Total in SLOC				1000

OK Cancel Help

Driver de costos para la iteración V:

EAF - Mod consult y repos

base + incr % = rating

	RCPX	RUSE	PDIF	PERS	PREX	FCIL	USR1	USR2
base	LO	LO	NOM	LO	LO	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%	0%	0%	0%

EAF is also affected by Schedule

EAF: 1.11

OK Cancel Help

SLOC Input Dialog - Mod consult y repos

Sizing Method

☐ SLOC

☒ Function Points

☐ Adaptation and Reuse

Breakage

% of code thrown away due to requirements evolution and volatility

REVL 0.00

Module Size in Function Points

Language JAVA Change Multiplier 40

Function Type	# of Function Points			SubTotal
	Low	Average	High	
Internal Logical Files	0	0	1	15
External Interface Files	0	0	0	0
External Inputs	1	0	0	3
External Outputs	1	0	0	4
External Inquiries	1	0	0	3
Total Unadjusted Function Points				25
Equivalent Total in SLOC				1000

OK Cancel Help

Driver de costos para la iteración VI:

EAF - Mod Mantenimiento

base + incr % = rating

	RCPX	RUSE	PDIF	PERS	PREX	FCIL	USR1	USR2
base	LO	LO	NOM	LO	LO	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%	0%	0%	0%

EAF is also affected by Schedule

EAF: 1.11

OK Cancel Help

SLOC Input Dialog - Mod Mantenimiento

Sizing Method

☐ SLOC

☒ Function Points

☐ Adaptation and Reuse

Breakage

% of code thrown away due to requirements evolution and volatility

REVL 0.00

Module Size in Function Points

Language JAVA Change Multiplier 40

Function Type	# of Function Points			SubTotal
	Low	Average	High	
Internal Logical Files	0	0	1	15
External Interface Files	0	0	0	0
External Inputs	1	0	0	3
External Outputs	1	0	0	4
External Inquiries	1	0	0	3
Total Unadjusted Function Points				25
Equivalent Total in SLOC				1000

OK Cancel Help

Nomenclatura para la interpretación de los resultados:

Ched: cuantas semanas me va a llevar desarrollar mi producto software.

Cost: costos reales del esfuerzo del producto software.

Staff: cuantas personas se necesitan para el desarrollo del producto software.

Early desing: aun no se tiene un diseño determinado. En esta etapa los costos suben debido que aun no se cuenta con variables importantes.

Project Name: Adm Hotelera

Scale Factor

Schedule

Development Model: Early Design

X	Module Name	Module Size	LABOR Rate (\$/month)	EAF	Language	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	Mod seguridad	F:1000	6000.00	0.57	JAVA	3.9	2.2	449.6	13344.81	13.3	0.2	0.0
	Mod Administra	F:2080	6000.00	2.22	JAVA	8.1	18.1	115.1	108384.17	52.1	1.5	0.0
	Mod Reservas	F:1360	6000.00	1.11	JAVA	5.3	5.9	230.1	35468.47	26.1	0.5	0.0
	Mod Gestion de	F:1000	6000.00	1.11	JAVA	3.9	4.3	230.1	26079.76	26.1	0.4	0.0
	Mod consult y	F:1000	6000.00	1.11	JAVA	3.9	4.3	230.1	26079.76	26.1	0.4	0.0
	Mod Mantenimie	F:1000	6000.00	1.11	JAVA	3.9	4.3	230.1	26079.76	26.1	0.4	0.0

Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK
Optimistic	26.3	10.7	283.0	157742.61	21.2	2.5	
Most Likely	39.2	12.2	189.6	235436.72	31.6	3.2	0.0
Pessimistic	58.9	13.9	126.4	353155.09	47.5	4.2	

Total Lines of Code:

7440

Estimación del esfuerzo en forma manual:

Precedentedness	6.20
Development Flexibility	3.04
Architecture / risk resolution	2.83
Team cohesion	3.29
Process maturity	7.80

$$\sum fe(i) = 6.20 + 3.04 + 2.83 + 3.29 + 7.8 = 23.16$$

$$E(\text{sin ajustar}) = a * (\text{tamaño})^{\text{factores de escala}}$$

$$E(\text{sin ajustar}) = 2.94 * (\text{KLOC})^{(0.65 + 0.01 * \sum fe(i))}$$

$$E(\text{sin ajustar}) = 2.94 * 9^{(0.65 + 0.01 * 23.16)} = 6,13$$

Estimación del esfuerzo para la iteración I:

$$PFSA = 25$$

$$EM_i = 0.98 * 0.95 * 1 * 0.83 * 0.87 * 1 * 1 * 1 = 0.57$$

$$\text{KLOC} = (25 \times 46) / 1000 = 1,15$$

$$Ea = 1,15 \times 0,57 = 0,655$$

Estimación del esfuerzo para la iteración II:

$$\text{PFSA} = 52$$

$$\text{II } EM_i = 0,98 \times 0,95 \times 1 \times 1,62 \times 1,33 \times 1 \times 1 \times 1 = 2,22$$

$$\text{KLOC} = (52 \times 46) / 1000 = 2,392$$

$$Ea = 2,392 \times 2,22 = 5,31$$

Estimación del esfuerzo para la iteración III:

$$\text{PFSA} = 34$$

$$\text{II } EM_i = 0,98 \times 0,95 \times 1 \times 1,26 \times 1,12 \times 1 \times 1 \times 1 = 1,11$$

$$\text{KLOC} = (34 \times 46) / 1000 = 1,564$$

$$Ea = 1,564 \times 1,11 = 1,73$$

Estimación del esfuerzo para la iteración IV:

$$\text{PFSA} = 25$$

$$\text{II } EM_i = 0,98 \times 0,95 \times 1 \times 1,26 \times 1,12 \times 1 \times 1 \times 1 = 1,11$$

$$\text{KLOC} = (25 \times 46) / 1000 = 1,15$$

$$Ea = 1,15 \times 1,11 = 1,27$$

Estimación del esfuerzo para la iteración V:

$$\text{PFSA} = 25$$

$$\text{II } EM_i = 0,98 \times 0,95 \times 1 \times 1,26 \times 1,12 \times 1 \times 1 \times 1 = 1,11$$

$$\text{KLOC} = (25 \times 46) / 1000 = 1,15$$

$$Ea = 1,15 \times 1,11 = 1,27$$

Estimación del esfuerzo para la iteración VI:

$$\text{PFSA} = 25$$

$$\text{II } EM_i = 0,98 \times 0,95 \times 1 \times 1,26 \times 1,12 \times 1 \times 1 \times 1 = 1,11$$

$$\text{KLOC} = (25 \times 46) / 1000 = 1,15$$

$$Ea=1,15 \times 1,11=1,27$$

$$Ea(\text{total})= 0,655+5,31+1,73+1,27+1,27+1,27= 11,5$$

Solución:**Cuadro de comparación del esfuerzo**

Tabla de asignación de los resultados obtenidos para el tipo 1 (Orgánico), Tipo 2 (modelo cocomo) y Tipo 3 (manual).

Esfuerzo Total	
Tipo 1	35,14
Tipo 2	39,2
Tipo 3	11,5

Planificación del Proyecto

A partir del Documento de Especificación de Requisitos, de la distribución de actividades elaborado con el Mapa de Actividades, utilizando los resultados de la estimación del tamaño y del esfuerzo requeridos para el proyecto calculados con el modelo COCOMOII, se deberá realizar un plan de trabajo que refleje el conjunto de actividades que se realizará, incluyendo la estimación del esfuerzo por cada actividad y ciclo del proyecto, así como el cálculo del costo total.

La Planificación debe realizarse utilizando una herramienta que permita distribuir el esfuerzo y visualizar gráficamente la distribución de tareas, roles y ciclos del proyecto. De acuerdo a la estimación de esfuerzo obtenida mediante la aplicación del método de COCOMO II, a continuación realizaremos la planificación de horas y costos para nuestro Proyecto.

Partiendo de:

Esfuerzo aplicando COCOMO
II
Esfuerzo
= 39,2 meses

Calculamos la duración del Proyecto:

Duración = Cant. Meses * 8 hs * 20 días
Duración = 6272 hs

Una vez que obtenemos la duración, como siguiente paso realizamos la planificación de horas tanto para las iteraciones como para los incrementos que componen cada una.

Iteraciones	Porcentaje de hs por iteración	Horas por iteración	Incrementos	Porcentaje de horas por incremento	Horas por incremento	Fase	Porcentajes de horas por fase de incremento	Horas por fase de incremento	Días por Iteración	Días por Incremento		
Iteración 1	20%	1.254										
			Incremento 1	50%	627	Gestión	10%	63	157	78		
						Análisis	15%	94				
						Diseño	15%	94				
						Desarrollo	40%	251				
						Pruebas	20%	125				
			Incremento 2	50%	627	Gestión	10%	63		78		
						Análisis	15%	94				
						Diseño	15%	94				
						Desarrollo	40%	251				
						Pruebas	20%	125				
Iteración 2	20%	1.254										
			Incremento 1	35%	439	Gestión	10%	44	157	55		
						Análisis	15%	66				
						Diseño	15%	66				
						Desarrollo	40%	176				
						Pruebas	20%	88				
			Incremento 2	30%	376	Gestión	10%	38		47		
						Análisis	15%	56				
						Diseño	15%	56				
						Desarrollo	40%	151				
						Pruebas	20%	75				

			Incremento 3	35%	439	Gestión	10%	44			55		
						Análisis	15%	66					
						Diseño	15%	66					
						Desarrollo	40%	176					
						Pruebas	20%	88					
Iteración 3	15%	941										118	
			Incremento 1	35%	329	Gestión	10%	33		41			
						Análisis	15%	49					
						Diseño	15%	49					
						Desarrollo	40%	132					
						Pruebas	20%	66					
										35			
			Incremento 2	30%	282	Gestión	10%	28					
						Análisis	15%	42					
						Diseño	15%	42					
						Desarrollo	40%	113					
						Pruebas	20%	56					
										41			
			Incremento 3	35%	329	Gestión	10%	33					
						Análisis	15%	49					
						Diseño	15%	49					
						Desarrollo	40%	132					
						Pruebas	20%	66					
Iteración 4	20%	1.254										157	55
			Incremento 1	35%	439	Gestión	10%	44					
						Análisis	15%	66					
						Diseño	15%	66					
						Desarrollo	40%	176					

						Pruebas	20%	88			
			Incremento 2	30%	376	Gestión	10%	38		47	
						Análisis	15%	56			
						Diseño	15%	56			
						Desarrollo	40%	151			
						Pruebas	20%	75			
			Incremento 3	35%	439	Gestión	10%	44		55	
						Análisis	15%	66			
						Diseño	15%	66			
						Desarrollo	40%	176			
Pruebas	20%	88									

Iteración 5	10%	627							78	
			Incremento 1	50%	314	Gestión	10%	31		39
						Análisis	15%	47		
						Diseño	15%	47		
						Desarrollo	40%	125		
						Pruebas	20%	63		
			Incremento 2	50%	314	Gestión	10%	31		39
						Análisis	15%	47		
						Diseño	15%	47		
						Desarrollo	40%	125		
						Pruebas	20%	63		

Iteración 6	15%	941							118	47
			Incremento 1	40%	376	Gestión	10%	38		
						Análisis	15%	56		
						Diseño	15%	56		

						Desarrollo	40%	151		
						Pruebas	20%	75		
			Incremento 2	30%	282	Gestión	10%	28		
						Análisis	15%	42		
						Diseño	15%	42		
						Desarrollo	40%	113		
						Pruebas	20%	56		
			Incremento 3	30%	282	Gestión	10%	28		
						Análisis	15%	42		
						Diseño	15%	42		
						Desarrollo	40%	113		
						Pruebas	20%	56		

Definimos los recursos necesarios para llevar adelante nuestro Proyecto (Equipo de trabajo) junto con el costo de cada uno.

Recursos	Costo x mes	Costo x hora
Jefe de Proyecto	\$ 30.000	\$ 188
Analista Senior	\$ 20.000	\$ 125
Diseñador Senior	\$ 19.000	\$ 119
Programador Senior	\$ 17.000	\$ 106
Programador Semi Senior	\$ 12.000	\$ 75
Programador Junior	\$ 8.000	\$ 50
Tester	\$ 7.000	\$ 44

Finalizada la planificación detallada anteriormente, utilizamos la herramienta Microsoft Project para realizar la asignación de recursos a las tareas, administrar el presupuesto, analizar las cargas de trabajo y dar seguimiento al progreso del Proyecto.

Distribución de fechas

	Task	Nombre de tarea	Work	Duration	Start	Finish
1		Proyecto SIGHO	6.272 hours	784 days	Mon 02/06/14	Thu 01/06/17
2		Iteración 1	1.254 hours	157 days	Mon 02/06/14	Tue 06/01/15
3		Incremento 1	627 hours	78 days	Mon 02/06/14	Wed 17/09/14
46		Incremento 2	627 hours	78 days	Thu 18/09/14	Mon 05/01/15
68		Iteración 2	1.256 hours	157 days	Tue 06/01/15	Wed 12/08/15
69		Incremento 1	440 hours	55 days	Tue 06/01/15	Mon 23/03/15
95		Incremento 2	376 hours	47 days	Tue 24/03/15	Wed 27/05/15
117		Incremento 3	440 hours	55 days	Tue 28/05/15	Wed 12/08/15
138		Iteración 3	939 hours	118 days	Thu 13/08/15	Mon 25/01/16
139		Incremento 1	329 hours	41 days	Thu 13/08/15	Thu 08/10/15
164		Incremento 2	281 hours	35 days	Fri 09/10/15	Thu 26/11/15
184		Incremento 3	329 hours	41 days	Fri 27/11/15	Fri 22/01/16
206		Iteración 4	1.256 hours	157 days	Tue 26/01/16	Wed 31/08/16
207		Incremento 1	440 hours	55 days	Tue 26/01/16	Mon 11/04/16
233		Incremento 2	376 hours	47 days	Tue 12/04/16	Wed 15/06/16
254		Incremento 3	440 hours	55 days	Thu 16/06/16	Wed 31/08/16
276		Iteración 5	628 hours	78 days	Thu 01/09/16	Mon 19/12/16
277		Incremento 1	315 hours	39 days	Thu 01/09/16	Tue 25/10/16
302		Incremento 2	313 hours	39 days	Wed 26/10/16	Mon 19/12/16
324		Iteración 6	939 hours	118 days	Tue 20/12/16	Thu 01/06/17
325		Incremento 1	377 hours	47 days	Tue 20/12/16	Wed 22/02/17
351		Incremento 2	281 hours	35 days	Thu 23/02/17	Wed 12/04/17
373		Incremento 3	281 hours	35 days	Thu 13/04/17	Wed 31/05/17

Costo total

	Task Name	Total Cost	Baseline	Variance	Actual	Remaining
1	Proyecto SIGHO	\$ 591.706,96	\$ 0,00	\$ 591.706,96	\$ 400,00	\$ 591.306,96
2	Iteración 1	\$ 121.009,95	\$ 0,00	\$ 121.009,95	\$ 0,00	\$ 121.009,95
68	Iteración 2	\$ 114.199,00	\$ 0,00	\$ 114.199,00	\$ 400,00	\$ 113.799,00
138	Iteración 3	\$ 89.938,00	\$ 0,00	\$ 89.938,00	\$ 0,00	\$ 89.938,00
206	Iteración 4	\$ 119.755,00	\$ 0,00	\$ 119.755,00	\$ 0,00	\$ 119.755,00
276	Iteración 5	\$ 58.759,00	\$ 0,00	\$ 58.759,00	\$ 0,00	\$ 58.759,00
324	Iteración 6	\$ 88.046,00	\$ 0,00	\$ 88.046,00	\$ 0,00	\$ 88.046,00

El Proyecto SISHO tendrá una duración de 3 años a un costo de aproximadamente \$ 600.000

Gestion de riesgos

A partir del Documento de Especificación de Requisitos:

1. Identificar **10** posibles Riesgos (del Proyecto, Técnicos, del Negocio)
2. Estimar el impacto de cada uno con valores de 0 a 5, siendo 0 el menor impacto esperado y 5 el máximo valor de impacto esperado.
3. Estimar la probabilidad de ocurrencia de cada uno con valores de 0 a 5, siendo 0 la menor probabilidad esperada y 5 el máximo valor de probabilidad de ocurrencia.
4. Multiplicar los valores y determinar los 5 factores de Riesgo más altos.
5. Definir un Plan de Gestión del Riesgo que contemple las acciones a tomar para reducir cada uno de los riesgos identificados para el Proyecto.

	Titulo	Tipo	Descripcion	Identifica do Por	Fecha de Identificaci on	Responsa ble	Estado	Clasificaci on	Accion de Mitigacion	Impac to	Probabilid ad de Ocurrenci a	Contingen cia	Priorid ad	Impac to Costo	Impac to Calida d	Impacto Cronogra ma	Exposici on al Riesgo
1	Entrega tardia de elementos de hardware	Riesgo	Posibilidad de retraso en la entrega del hardware necesario para la implementacion del sistema en el/los hoteles correspondientes debido a restricciones de importacion	Gerente IT	07/05/2014	Gerente IT	Asignado	Riesgo de Proyecto	Consulta a diferentes proveedores y establecimiento de contrato para la entrega	3	4	Utilizacion de Hardware disponible hasta la entrega del nuevo, sacando importancia temporal al requisito de performance	3	4	3	2	36
2	Renuncia de integrante del equipo de desarrollo	Riesgo	Posibilidad de que algun integrante del equipo de desarrollo renuncie una vez comenzado el proyecto	Jefe de Desarrollo	08/05/2014	Jefe de Desarrollo	Asignado	Riesgo de Proyecto	Realizar tareas de a pares	3	1	Solicitar aviso a los empleados con bastante antelacion, de manera de poder suplir la ausencia	1	2	3	3	8
3	Volumen de operacion del software mayor al esperado	Riesgo	La disponibilidad web del software y las oportunidades de negocio presentadas al cliente hace que este aumente su visibilidad, y por tanto, las consultas/reservas que el sistema soporta	Jefe de Desarrollo	15/05/2014	Jefe de Desarrollo	Asignado	Riesgo Tecnico	Sobredimensionar la performance del software en un 20%	2	3	Comunicacion con el cliente de la posibilidad de incremento en el negocio, y los requisitos especificados	2	3	3	1	21

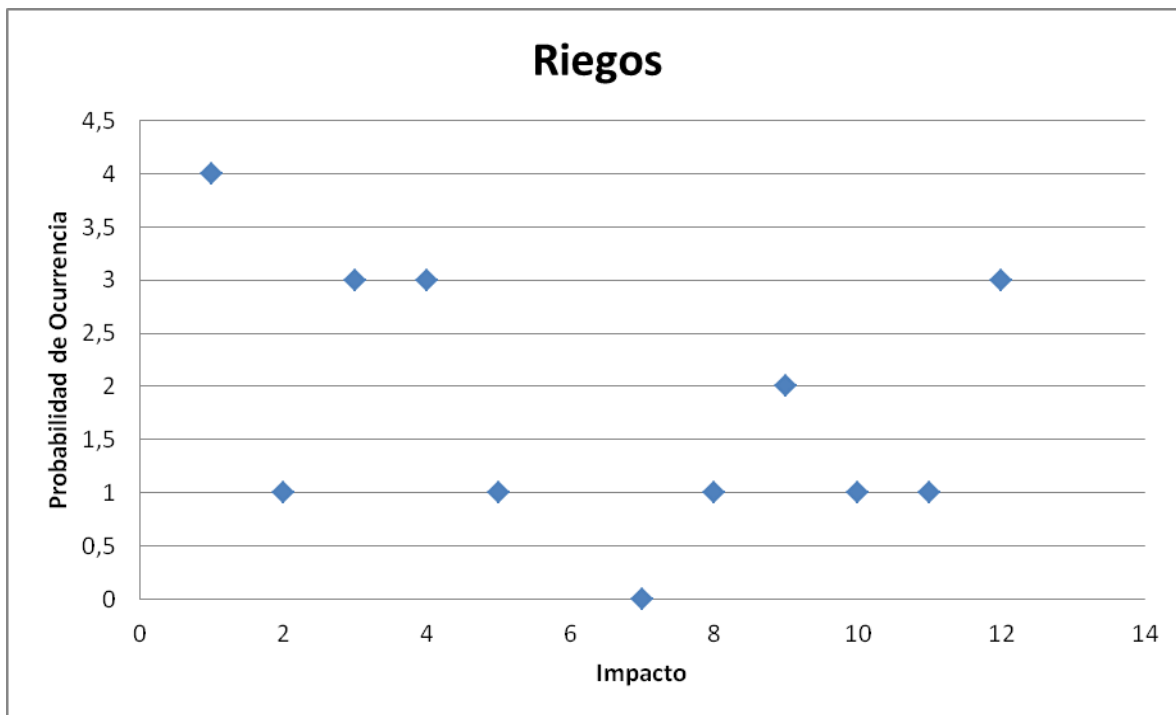
Documento Integrador

4	Solicitud de realizaci on con pantallas táctiles	Riesg o	El cliente quedó tan interesado en la oportunidad de que el software se manejara con pantallas táctiles que lo transforma en un requisito	Analista	08/05/2014	Jefe de Desarrollo	Asigna do	Riesgo Tecnico	Interfaces deberan desarrollarse para dispositivos táctiles	3	3	Portar la aplicacion para dispositivos móviles	3	3	1	4	24
5	Diseño deficiente de la base de datos	Riesg o	Las tablas consideradas y las relaciones entre las mismas hace que las consultas sean deficientes y tomen mas tiempo del solicitado	Jefe de Desarrollo	22/05/2014	Analista funcional	Asigna do	Riesgo Tecnico	Incremento en la asignacion de recursos hardware para compensar los tiempos de respuesta	2	1	Rediseño de la base de datos	1	2	4	1	7

ID	Título	Tipo	Descripción	Identificado Por	Fecha de Identificación	Responsable	Estado	Clasificación	Acción de Mitigación	Impacto	Probabilidad de Ocurrencia	Contingencia	Prioridad	Impacto Costo	Impacto Calidad	Impacto Cronograma	Exposición al Riesgo
6	Especialización del software innecesaria	Riesgo	El desarrollo esta tan ligado al cliente que limita las posibilidades de venta a clientes similares	Analista	22/05/2014	Analista	Asignado	Riesgo de Negocio	Estudiar la reusabilidad del código e interfaces	3	1	Incrementar la abstracción y proceder desde allí a la especificación	3	4	4	1	9
7	Ciclo de vida de software incorrecto	Riesgo	Dada la evaluación que se hizo al comienzo sobre los ciclos de vida de software a adoptar arrojó valores cercanos entre dos posibilidades, se evidencia que la elección fue errónea	Analista	07/05/2014	Analista	Asignado	Riesgo de Proyecto	Planificar una holgura para mitigar los retrasos	3	2	Plantear el cambio temprano del ciclo de vida del software elegido	4	2	4	4	20
8	Salida de personal de gerencia	Riesgo	Posibilidad de renuncia del personal de gerencia o dirección del proyecto, por razones diversas	Project Manager	21/05/2014	Project Manager	Asignado	Riesgo de Negocio	Hacer que personal inmediatamente dependiente este al corriente del proyecto	4	1	Disponer de recursos para ofrecer al personal que se retira, y solicitar tiempo suficiente de anticipación para conseguir un reemplazo correspondiente	4	4	4	3	11
9	Dimensión del proyecto superior a lo esperado	Riesgo	La dimensión del proyecto calculada difiere de la observada a medida se desarrolla el proyecto	Project Manager	31/05/2014	Project Manager	Asignado	Riesgo de Proyecto	Planificar una holgura para mitigar los retrasos	4	1	Solicitar extensión de plazo de entrega al cliente	3	3	4	4	11

Documento Integrador

10	Experiencia insuficiente en el negocio	Riesgo	El relevamiento de los requisitos omite datos importantes sobre el desenvolvimiento del negocio	Project Manager	31/05/2014	Analista	Asignado	Riesgo de Proyecto	Mantener comunicación fluida y testeo constante con el cliente y usuario final	4	3	Solicitar personal del cliente en el equipo de desarrollo y testeo	3	4	5	3	36
----	--	--------	---	-----------------	------------	----------	----------	--------------------	--	---	---	--	---	---	---	---	----



Utilizando la fórmula de exposición al riesgo, se establece que los siguientes son los riesgos más altos:

- 1- Entrega tardía de elementos de hardware
- 2- Experiencia insuficiente en el negocio
- 3- Solicitud de realización del proyecto con pantallas táctiles
- 4- Volumen de operación del software mayor al esperado
- 5- Ciclo de vida del software incorrecto

Gestión de Configuración

Las Actividades de Gestión de Configuración constituyen un apartado dentro del Plan de Gestión de Configuración, éste es un documento se debe producir al comienzo de cada proyecto y que define las políticas, estándares y procedimientos que se van a utilizar para gestionar la configuración en el transcurso de dicho proyecto.

A partir del Documento de Especificación de Requisitos definir:

1. Identificación de la configuración

Identificar y asignar nombres significativos y consistentes a todos y cada uno de los elementos que forman parte del producto software, a cada ECS. Es importante tener en cuenta los productos definidos en el Mapa de Actividades y en la Planificación del Proyecto.

2. Definir líneas base

Definir las líneas base del proyecto e identificar cuales van a ser los ECS que contenga cada línea base.

3. Establecer relaciones

Establecer las relaciones que se darán entre los ECS.

4. Control de cambios de la configuración

Proporcionar un mecanismo formal para controlar los cambios de cada uno de los elementos de configuración. Definir el formato de la solicitud de cambio, quienes pueden solicitarlo y quienes pueden aprobarlo.

5. Generación de informes de estado

Describir 3 informes que nos debería proporcionar nuestro sistema de gestión de configuración. Detallar los campos que contendrá cada informe como así también quién tendrá permisos para acceder a ellos.

6. Auditoría de la configuración

Determinar los 5 puntos críticos que controlarían en el caso de estar realizando una auditoria externa al proceso de Gestión de Configuración.

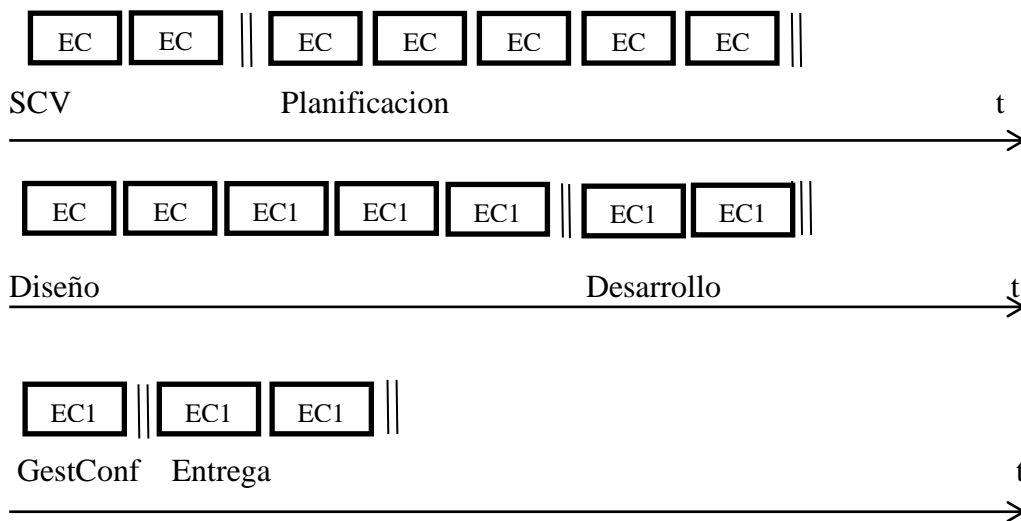
1. Identificación de la configuración

Nombre significativo del elemento de configuración:

PROYECTO_LINEABASE_FASE_NOMBRE_TIPO_VERSION

EC	Elementos del Producto Software	Nombre del Elemento de configuración
1	Especificacion de Requisitos	SISHO_SCV_AnaReq_EspecRequisitos_DOC_V1.0
2	Selección del Modelo de Ciclo de Vida	SISHO_SCV_SMCV_SeleccionMCV_DOC_V1.0
3	Mapa de Actividades	SISHO_Planificacion_MapaAct_MapaActividades_DOC_V1.0
4	Planificación de tareas	SISHO_Planificacion_PlanTar_PlanifTareas_DOC_V1.0
5	Documento Analisis de Riesgos	SISHO_Planificacion_Riesgos_AnalisisRiesgos_DOC_V1.0
6	Documento de Calidad del Producto	SISHO_Planificacion_GestionCalidad_Calidad_DOC_V1.0
7	Estudio de Viabilidad	SISHO_Planificacion_PreDesarrollo_EstViabilidad_DOC_V1.0
8	Documentación de Casos de Uso	SISHO_Diseño_Diseño_CasosDeUso_DOC_V1.0
9	Documento de Arquitectura	SISHO_Diseño_Diseño_Arquitectura_DOC_V1.0
10	Diagrama de Componentes	SISHO_Diseño_Diseño_DiagComponentes_DOC_V1.0
11	Diagrama de Despliegue	SISHO_Diseño_Diseño_DiagDespliegue_DOC_V1.0
12	Diagrama de Clases	SISHO_Diseño_Diseño_DiagClases_DOC_V1.0
13	Archivos de Código Fuente	SISHO_Desarrollo_Desarrollo_<ClasesContenidas>.<ext>
14	Documentación de Pruebas	SISHO_Desarrollo_PostDesarrollo_Pruebas_V1.0
15	Documento de Especificación de Gestión de Configuración	SISHO_GestConf_Configuracion_GestionDeConfiguracion_V1.0
16	Manual de Operación	SISHO_Entrega_Operacion_ManualUsuario_V1.0
17	Manual de Formación	SISHO_Entrega_Operacion_ManualFormacion_V1.0

2. Definir líneas base



3. Establecer relaciones

4. Control de cambios de la configuración

Mecanismo formal para la solicitud de cambios sobre Elementos de configuración:

- 1- Realizar un análisis sobre el impacto que tendrá el cambio
- 2- Realizar una solicitud de cambio utilizando el template
- 3- Adjuntar a la solicitud de cambio el análisis realizado en el punto 1
- 4- Realizar revisión del análisis y la solicitud de cambio por parte de la Comisión de Control de Cambios
- 5- Se realiza aprobación o rechazo del cambio

Template para Solicitud de Cambio:

Numero de Cambio:	SCEC-0000
Fecha Solicitud:	AAAA-MM-DD
Solicitante:	Apellido y Nombre
Responsable:	Apellido y Nombre
Sector:	Sector
Motivo:	
Aceptado:	Si/No
Observaciones:	Breve descripción del cambio y sus implicaciones

Quien puede solicitar cambios:

Cualquier integrante del equipo que trabaja en el proyecto

Template para Control de Cambio:

Numero de Cambio:	CCEC-0000
Elemento de Configuración involucrado:	IdEC
Fecha de Realización de Cambio:	AAAA-MM-DD
Responsable del cambio:	Apellido y Nombre
Responsable de revisión del cambio:	Apellido y Nombre
Fecha de revisión:	AAAA-MM-DD
Observaciones:	

Comisión de Control de Cambio:

Es el grupo encargado de aprobar y controlar las solicitudes y cambios de Elementos e configuración. Estará integrada por el líder del proyecto y los jefes de cada equipo involucrado en el proyecto.

5. Generación de informes de estado

A) Informe de elementos de configuracion: detalle de los elementos de configuracion que componen el sistema

Campos del informe: <Numero><IDEC><Version><Link al Documento>

B) Informe de solicitud de cambios: detalle de los cambios solicitados sobre los EC

Campos del informe: <Numero><IDEC><Solicitante><Fecha Solicitud><Estado>

C) Informe control de cambios: detalle de lo controles realizados sobre los cambios realizados en base a las solicitudes de cambio de EC

Campos del Informe: <Numero><IDEC><Fecha de Control><Responsable><Estado>

6. Auditoría de la configuración

- 1- Registro de los elementos de configuración de los proyectos
- 2- Importancia y fiabilidad del proceso seguido para la solicitud y realización de cambios en los elementos de configuración de los proyectos
- 3- Histórico de las solicitudes de cambios sobre los elementos de configuración y su realización
- 4- Revisión y comprobación del proceso de gestión de configuración, y sus subprocesos
- 5- Comparación de los hallazgos con los resultados de auditorías internas, y si hubo mejoras a lo largo del tiempo en dichos hallazgos

Modelos de calidad de producto

El siguiente apartado contiene un detalle de Factores y Criterios del Modelo de Calidad de McCall. El objetivo es, a partir de la Especificación de Requisitos, seleccionar los Factores, Criterios y Métricas mas adecuados para aplicar en el desarrollo del software especificado.

A) Analizar los 11 factores en función a los siguientes puntos

- Relación entre los factores. Ejemplo: La facilidad de uso incide negativamente con el rendimiento
- Relacionar el costo beneficio. Ejemplo: Corrección tiene un beneficio muy alto más allá del costo
- Importancia para nuestro proyecto. Ejemplo: ciclo de vida largo, necesito facilidad de mantenimiento.

B) Seleccionar los 4 factores más relevantes en función al análisis anterior.

C) Para cada uno de los criterios relacionados a los 4 factores seleccionados, crear dos métricas con el formato del siguiente ejemplo:

Métricas de Facilidad de Uso

Facilidad de Operación

Nombre: Pasos por Acción

Objetivo: Comprobar la facilidad con que se opera (Fácil de operación)

Método de Aplicación: Se considerarán los pasos (P) de cada acción simples (A) del flujo normal de cada caso de uso (CU).

Fórmula: P para cada A.

Fuente de medición: Casos de Uso, Pantalla

Rango de Aceptación: $0 < P < 5$

Modelo de McCall

Perspectiva de los 11 Factores de Calidad

PUNTO DE VISTA	FACTORES
Operación del producto	<ul style="list-style-type: none"> - Facilidad de uso - Integridad - Corrección - Fiabilidad - Eficiencia
Revisión del producto	<ul style="list-style-type: none"> - Facilidad de mantenimiento - Facilidad de prueba - Flexibilidad
Transición del producto	<ul style="list-style-type: none"> - Facilidad de reutilización - Interoperabilidad - Portabilidad

Dependencia entre los 11 Factores de Calidad

	CORRECCION									
	FIABILIDAD									
	EFICIENCIA									
	INTEGRIDAD									
	FACILIDAD DE USO									
	FACIL. DE MANTEN.									
	FACIL. DE PRUEBA									
	FLEXIBILIDAD									
	PORTABILIDAD									
	REUSABILIDAD									
	INTEROPERABILIDAD									
CORRECCION										
FIABILIDAD	○									
EFICIENCIA										
INTEGRIDAD				●						
FACILIDAD DE USO	○	○	●	○						
FACIL. DE MANTEN.	○	○	●		○					
FACIL. DE PRUEBA	○	○	●		○	○				
FLEXIBILIDAD	○	○	●	●	○	○	○			
PORTABILIDAD			●			○	○			
REUSABILIDAD		●	●	●		○	○	○	○	
INTEROPERABILIDAD			●	●				○		

Cuando hay un alto grado de calidad para un factor, ¿qué grado de calidad se puede esperar para otros factores?

○ Alto

● Bajo

Relación de Factores con Criterios del Modelo McCall

FACTOR	CRITERIOS	METRICAS
Facilidad de uso	<ul style="list-style-type: none"> - Facilidad de operación - Facilidad de comunicación - Facilidad de aprendizaje 	
Integridad	<ul style="list-style-type: none"> - Control de accesos - Facilidad de auditoría 	
Corrección	<ul style="list-style-type: none"> - Completitud - Consistencia - Trazabilidad 	
Fiabilidad	<ul style="list-style-type: none"> - Precisión - Consistencia - Tolerancia a fallos - Modularidad - Simplicidad 	
Eficiencia	<ul style="list-style-type: none"> - Eficiencia en ejecución - Eficiencia en almacenamiento 	
Facilidad de mantenimiento	<ul style="list-style-type: none"> - Modularidad - Simplicidad - Consistencia - Concisión - Auto descripción 	
Facilidad de prueba	<ul style="list-style-type: none"> - Modularidad - Simplicidad - Auto descripción - Instrumentación 	
Flexibilidad	<ul style="list-style-type: none"> - Auto descripción - Capacidad de expansión - Generalidad - Modularidad 	
Reusabilidad	<ul style="list-style-type: none"> - Auto descripción - Generalidad - Modularidad - Independencia entre sistema y software - Independencia del hardware 	
Interoperabilidad	<ul style="list-style-type: none"> - Modularidad - Compatibilidad de comunicaciones - Compatibilidad de datos 	
Portabilidad	<ul style="list-style-type: none"> - Auto descripción - Modularidad - Independencia entre sistema y software - Independencia del hardware 	

FACTOR	RELACION ENTRE FACTORES	COSTO - BENEFICIO	IMPORTANCIA PARA EL PROYECTO
Facilidad de uso	Incide negativamente con eficiencia y rendimiento. Incide positivamente con facilidad de mantenimiento flexibilidad y facilidad de prueba.	Costo :Medio Hacer un producto más orientado a la comodidad del usuario para que sea mas fácil la capacitación y el uso del sistema	Este atributo es muy relevante para cualquier proyecto
Integridad	Incide negativamente con eficiencia e interoperatividad. Incide positivamente con facilidad de uso.	Costo: bajo Verificar que el producto haga correctamente su función. Beneficio: Seguridad que el producto funciona bien.	Muy Alta
Corrección	Incide positivamente con Fiabilidad, facilidad de uso, facilidad de mantenimiento, facilidad de prueba y flexibilidad.	Costo =Alto. Beneficio alto más allá de su costo	Moderada, en función de los errores que se cometan dentro de la fase que se encuentran.
Fiabilidad	Incide negativamente con Reusabilidad Incide positivamente con facilidad de uso, facilidad de mantenimiento, facilidad de prueba y flexibilidad.	Costo =Alto. Maximizar beneficios del producto más allá del costo.	Atributo que debe (en lo posible) cubrir el100% del producto.
Eficiencia	Incide negativamente con interoperabilidad, integridad, facilidad de uso, facilidad de mantenimiento, facilidad de prueba, flexibilidad, portabilidad y reusabilidad	Costo=bajo Beneficio= visualizar la info. en el tiempo de respuesta requerido	Atributo que debe cubrirse en un 100%.
Facilidad de mantenimiento	Induce negativamente con eficiencia Incide positivamente con facilidad de prueba, flexibilidad, corrección y fiabilidad	Costo = Alto Beneficio=Re alizar un buen diseño para que el producto sea	Facilidad de mantenimiento.

		mantenible.	
Facilidad de prueba	Índice negativamente con eficiencia incide positivamente con corrección, fiabilidad, flexibilidad, portabilidad y reusabilidad	Costo=Alto Beneficio=Agilizar la corrección	Testear rápidamente los nuevos requisitos ó la corrección de errores.
Flexibilidad	Incide negativamente con eficiencia e integridad Incide positivamente con corrección, fiabilidad, facilidad de uso, facilidad de mantenimiento, facilidad de prueba, portabilidad y reusabilidad.	Costo=medio Beneficio=se adapta fácilmente a los cambios funcionales	Necesidad cambios a tener en cuenta para modificaciones futuras del software
Facilidad de reutilización	Incide negativamente con fiabilidad, eficiencia e integridad Incide Positivamente con facilidad de prueba, flexibilidad, portabilidad e interoperatividad	Costo=medio Beneficio=Utilizar el mismo componente en múltiples proyectos	Automatización y facilidad de implementación para el uso de recursos disponibles
Interoperabilidad	Incide negativamente con eficiencia e integridad Incide positivamente con reusabilidad.	Costo=bajo Beneficio=multiplataforma	Necesidad de acoplamiento e interacción con otros sistemas.
Portabilidad	Incide negativamente con eficiencia incide positivamente con facilidad de prueba, flexibilidad y Reusabilidad	Costo=medio Beneficio=Cambio de hardware hace de que todas formas mi producto funcione.	Necesidad de cambios en distintas arquitecturas físicas.

Seleccionar los 4 factores más relevantes en función al análisis anterior.

Corrección – Fiabilidad – Facilidad de prueba – Facilidad de Mantenimiento.

Porque tienen una relación Beneficio frente a costo Alto.

C) Para cada uno de los criterios relacionados a los 4 factores seleccionados, crear dos métricas con el formato del siguiente ejemplo:

Métricas de Facilidad de Uso

Facilidad de Operación

Nombre: Pasos por Acción

Objetivo: Comprobar la facilidad con que se opera (Fácil de operación)

Método de Aplicación: Se considerarán los pasos (P) de cada acción simples (A) del flujo normal de cada caso de uso (CU).

Fórmula: P para cada A.

Fuente de medición: Casos de Uso, Pantalla

Rango de Aceptación: $0 < P < 5$

C.1) Métrica I de Corrección

Nombre: completitud.

Objetivo: El grado en que se ha conseguido la total implementación de las funciones requeridas

Método de Aplicación: Testeo individual de cada Módulo (M) del sistema.

Fórmula: Objetivos (O) por M.

Fuente de medición: Vistas, módulos.

Rango de Aceptación:

5= Métrica satisfactoria.

5< Métrica insatisfecha.

C.2) Métrica II de Corrección

Nombre: consistencia.

Objetivo: El uso de un diseño uniforme de técnicas de documentación a lo largo del proyecto de desarrollo de software.

Método de Aplicación: Testeo de la documentación de gestión de Planificación de recursos y estimación de proyecto.

Fórmula: Objetivos (O) por documentos (D).

Fuente de medición: documentación.

Rango de Aceptación:

4= Métrica satisfactoria.

4<> Métrica insatisfecha.

C.3) Métrica I de Fiabilidad

Nombre: Exactitud.

Objetivo: La precisión de los cálculos y el control.

Método de Aplicación: Testeo individual de cada función de calculo (F) del sistema.

Fórmula: Errores (E) para cada F.

Fuente de medición: funciones de cálculo, base de datos internas.

Rango de Aceptación:

4= Métrica satisfactoria.

4<> Métrica insatisfecha.

C.4) Métrica II de Fiabilidad

Nombre: Eficiencia en la ejecución.

Objetivo: El rendimiento en tiempo de ejecución de un programa.

Método de Aplicación: Testeo individual de los tiempos de ejecución (T) de cada Módulo del sistema.

Fórmula: Tiempo de ejecución para cada T.

Fuente de medición: módulos.

Rango de Aceptación:

4= Métrica satisfactoria.

4<> Métrica insatisfecha

C.5) Métrica I de Facilidad de Prueba

Nombre: Modularidad.

Objetivo: La independencia funcional de los componentes del programa.

Método de Aplicación: fan in/ fan out

Fórmula: O por cada CU

Fuente de medición: funciones y clases del sistema.

Rango de Aceptación:

4= Métrica satisfactoria.

4<> Métrica insatisfecha.

C.6) Métrica II de Facilidad de Prueba

Nombre: Instrumentación.

Objetivo El grado en que el programa muestra su propio funcionamiento e identifica errores que aparecen

Método de Aplicación: Errores (E) por cada ingreso/consulta (C).

Fórmula: E por C.

Fuente de medición: registros de consultas y validaciones de datos

Rango de Aceptación:

4= Métrica satisfactoria.

4<> Métrica insatisfecha.

C.7) Métrica I de Facilidad de Mantenimiento

Nombre: concisión.

Objetivo: Lo compacto que es el programa en términos de líneas de código.

Método de Aplicación: valores de entrada (A) y valores de salida (S).

Fórmula: A/S

Fuente de medición: módulos del sistema.

Rango de Aceptación:

4= Métrica satisfactoria.

4<> Métrica insatisfecha.

C.8) Métrica II de Facilidad de Mantenimiento

Nombre: auto-documentación

Objetivo: El grado en que el código fuente proporciona documentación significativa.

Método de Aplicación: Contar la cantidad de comentarios (P) de los módulos (M).

Fórmula: P por M.

Fuente de medición: módulos.

Rango de Aceptación:

4= Métrica satisfactoria.

4<> Métrica insatisfecha.

. Gestión de Calidad

A partir del Documento de Especificación de Requisitos diseñar y desarrollar algún procedimiento bajo el referencial de ISO 9001.

El equipo de software a trabajar en el desarrollo del producto cuenta con analistas, arquitectos y programadores que diseñan y construyen el producto. Asimismo, el proyecto requiere de un conjunto de tareas administrativas para gestionar las cuentas de clientes y proveedores, y un área de ventas que se ocupe de la comercialización del producto.

A partir de este contexto:

A) Diseñar los perfiles de cada integrante del equipo del proyecto, detallando las competencias y tareas de cada colaborador como así también la educación, formación, habilidades y experiencia requeridas.

B) Enumerar y detallar los recursos, la infraestructura y el ambiente de trabajo necesario para el proyecto, incluyendo las tareas administrativas y comerciales.

C) Definir detalladamente el proceso de producción indicando los puntos de control junto a sus condiciones de aprobación como así también los instrumentos, métodos y responsabilidades de cada miembro del equipo.

D) Indicar en cada caso las evidencias que se requerirán en las auditorias.

A) Diseñar los perfiles de cada integrante del equipo del proyecto, detallando las competencias y tareas de cada colaborador como así también la educación, formación, habilidades y experiencia requeridas.

Perfiles de cada puesto

Jefe de Proyecto

Educación	Título de Ingeniería en Informática ó Sistemas.
Formación	Diplomatura en Dirección de Proyectos. Máster en Informática.
Habilidades	<p>Liderazgo: por medio del desarrollo de una visión y una estrategia que será transferida al resto del equipo.</p> <p>Motivación: Estimular a los integrantes del equipo para que alcancen altos niveles de rendimiento y sepan afrontar cambios.</p> <p>Negociación y gestión de conflictos: Saber enfrentar situaciones conflictivas por medio del diálogo y elaborar soluciones.</p> <p>Resolución de problemas: Identificación de las causas, análisis de alternativas de solución, y toma de decisiones.</p>
Experiencia	<p>Mínimo 3 años de experiencia en el puesto.</p> <p>Experiencia en el desarrollo de sistemas de diversas envergaduras, utilizando diferentes metodologías y herramientas.</p> <p>Java Script senior.</p> <p>Manejo DB senior</p> <p>Diseño y arquitectura de sistemas</p> <p>Manejo de herramientas de planificación (Ej: MS Project)</p> <p>Manejo de herramientas de diseño (Ej: Enterprise Architect)</p>
Tareas	<p>Mantener coordinado todo el equipo, con comunicación fluida.</p> <p>Desarrollar el plan del Proyecto.</p> <p>Identificar los requerimientos y el alcance del Proyecto.</p> <p>Gestión del Proyecto.</p> <p>Realización de estimaciones.</p> <p>Asignación de recursos.</p> <p>Gestión de riesgos.</p>
Competencias	<p>Amplios conocimientos tecnológicos, variedad de lenguajes de programación, Bases de datos, Sistemas Operativos, Redes y Telecomunicaciones.</p> <p>Estar actualizado en cuanto a normas y regulaciones que tengan que ver con el desarrollo de software y productos de calidad</p> <p>Conocimientos de contrataciones.</p> <p>Desarrollo de nuevos productos.</p>

Analista Senior

Educación	Título de Analista de Sistemas ó título Analista Programador Universitario ó estudiante avanzado en Sistemas (como mínimo 25 materias aprobadas)
Formación	Conocimientos de Metodologías de Desarrollo de Sistemas y de Metodologías ágiles. Sólidos conocimientos de diagramación lógica, algoritmos y estructuras de datos. Sólidos conocimientos de programación, análisis y diseño tanto Estructurado como Orientado a Objetos Sólidos conocimientos de Sistemas de Información. Conocimientos de bases de datos relacionales
Habilidades	Muy buena comprensión, lectura y redacción Comunicación oral y escrita, especialmente con áreas no técnicas Metodología de trabajo Cumplimiento de plazos. Relaciones interpersonales y trabajo en equipo.
Experiencia	Mínimo 4 años de experiencia en el puesto. Manejo de herramientas de diseño.
Tareas	Armado de CU. Análisis de funciones. Planificación de la Gestión de la Configuración. Planificación de la documentación. Resguardo de registros. Definición y desarrollo de los requisitos del software.
Competencias	Actitud crítica, de perfeccionamiento y actualización permanente. Capacidad para trabajar en forma cooperativa y constructiva, en equipos Multidisciplinarios. Pro actividad. Flexibilidad. Capacidad de análisis y solución de problemas. Comunicación.

Diseñador Senior

Educación	Título de Analista de Sistemas ó título Analista Programador Universitario ó estudiante avanzado en Sistemas (como mínimo 25 materias aprobadas)
Formación	Conocimientos de metodologías ágiles. Conocimientos de diagramación lógica, algoritmos y estructuras de datos, programación, análisis y diseño. Conocimientos de patrones de diseño de software. Bases de datos.
Habilidades	Definir y diseñar la configuración de los componentes de las aplicaciones de acuerdo a la estructura del problema planteado, los requerimientos funcionales, los no funcionales y las necesidades del Proyecto.
Experiencia	Mínimo 4 años de experiencia en el puesto. Manejo de herramientas de diseño.
Tareas	Desarrollo de la arquitectura del sistema. Realización del diseño arquitectónico. Diseño de la base de datos. Diseño de interfaces.
Competencias	Creatividad.

	Planificación y control. Pensamiento de integración.
--	---

Programador Senior

Educación	Título de Ingeniería en Informática ó Sistemas. ó finalizando la carrera.
Formación	Conocimientos en programación estructurada y orientada a objetos. Lenguajes: Java /J2EE , C, C++. UML, Metodologías ágiles. Base de datos relacionales (MySQL).
Habilidades	Referente técnico dentro del equipo. Colaborar en definiciones arquitectónicas. Genera buen código y fácil de mantener. Cumple con los procesos, metodologías y estándares. No solamente recibe requerimientos, sino que los busca y genera.
Experiencia	Mínimo 5 años de experiencia.
Tareas	Reporta el estado y avance de sus tareas. Diseño de interfaces. ABM de determinados módulos del sistema. Validaciones Gestión de sesiones de usuarios. Asignaciones de datos. Cálculos. Generación de confirmaciones. Configuraciones
Competencias	Es bueno comunicando, pero principalmente escuchando. Puede participar en decisiones de alto nivel y colaborar si es necesario en actividades más operativas. Cumple con los plazos establecidos.

Programador Semi Senior

Educación	Título de Analista de Sistemas ó título Analista Programador Universitario ó finalizando la carrera.
Formación	Conocimientos de Java/J2EE. Base de datos relacionales (MySQL)
Habilidades	Se preocupa por aprovechar mejor su tiempo. Pide nuevas asignaciones cuando tiene tiempo disponible. Autosuficiente para llevar adelante una gran parte de sus tareas. Cumple con las metodologías y estándares
Experiencia	Mínimo 3 años de experiencia.
Tareas	Diseño de interfaces. ABM de determinados módulos del sistema. Validaciones Gestión de sesiones de usuarios. Modificaciones de estado. Registros. Cálculos.
Competencias	Buena comunicación Analítico Ordenado

	Capacidad de cambio
--	---------------------

Programador Junior

Educación	Estudiante inicial de la carrera de Sistemas (como mínimo 20 materias aprobadas)
Formación	Conocimientos de Java/J2EE
Habilidades	Entusiasmo por aprender y mejorar. Capacidad de cambio
Experiencia	Mínimo 1 años de experiencia.
Tareas	Consultas de determinados módulos del sistema. ABM de determinados módulos del sistema. Validaciones Registraciones
Competencias	Observador. Capacidad de concentración.

Tester

Educación	Estudiante inicial de la carrera de Sistemas (como mínimo 20 materias aprobadas)
Formación	Experiencia requerida en ejecución de testing Conocimiento de herramientas para la ejecución y seguimiento de testing. Conocimientos de SQL Conocimiento de diferentes Tipos de Testing (Volumen, stress, performance, funcional, técnico)
Habilidades	conocimientos de lógica necesaria para ejecutar pruebas técnicas, especialmente si se necesita utilizar herramientas de automatización Pensamiento crítico
Experiencia	Mínimo 1 año de experiencia.
Tareas	Planificar pruebas. Desarrollar las especificaciones de las pruebas. Ejecutar las pruebas
Competencias	Capacidad para planificar y formalizar Plan de Testing . Capacidad de Generación y Ejecución de Casos de Testing. Capacidad para orientar sus casos de prueba a comprobar cómo reacciona el producto ante picos de volumen, performance, stress, etc. Capacidad para orientar sus casos de prueba a corroborar si el producto se ejecuta funcionalmente en forma óptima

B) Enumerar y detallar los recursos, la infraestructura y el ambiente de trabajo necesario para el proyecto, incluyendo las tareas administrativas y comerciales.

Recursos

- Un escritorio y una silla ergonómica para cada integrante del equipo.
- Suministro eléctrico

- Conexión de red.
- Conexión a internet con 6Mb de banda ancha.
- Un teléfono interno para el equipo de desarrollo (programador Sr, Semi Sr y Jr), testing, analista, diseñador y jefe de proyecto.
- Una computadora de escritorio para el equipo de desarrollo y testing.
- Una notebook para el analista, diseñador y jefe de proyecto.
- Dos aires acondicionados ubicados en lados opuestos de la oficina.
- Un para base de datos y aplicación web.
- Elementos de networking (switch, routers, cableado).
- Conexión wifi.
- Tres matafuegos de clase A,B Y C y/o elementos contra incendios.
- Un sala de servidores.
- Un sala de desarrollo.
- Una sala de reuniones.
- Un baño.
- Un dispenser de agua para las salas de reuniones, y desarrollo.
- Aire acondicionado de Frío/Calor para las salas de desarrollo, reuniones y servidores.

Infraestructura

Aplicaciones y librerías de soporte para desarrollo:	Eclipse 3.0 J2EE 1.4 Sun Java System Application Server Platform Struts 1.3 jQuery 1.11 JSON Google Chrome Mozilla Firefox
Aplicaciones de soporte para testing	Selenium Oracle VM
Aplicaciones para análisis y diseño	Enterprise Architect 9.1
Base de datos	MySQL 9i MySQL Workbench 6.0
Sistemas operativo	Microsoft Windows 7 Ubuntu 13.04
Gestión, seguimiento, control de proyectos.	Web2Project 3.0 Pivotal tracker
Gestión de la configuración	SVN 1.8
Gestión de incidencias y bugs	JIRA 6.1
Herramientas de ofimática	Microsoft Office 2012 LibreOffice 4.2
Mensajería interna	Gmail

Ambiente de trabajo

- Nivel de luminosidad medio
- Ambiente climatizado
- Niveles de ruido controlados
- Disposición de los escritorios según roles.
- Ventanas en la oficina.

C) Definir detalladamente el proceso de producción indicando los puntos de control junto a sus condiciones de aprobación como así también los instrumentos, métodos y responsabilidades de cada miembro del equipo.

PROCESO DE PRODUCCIÓN

Análisis

Descripción	<ul style="list-style-type: none"> • Determinación de los requisitos del producto. • Obtención de la aprobación de los requerimientos por parte del cliente. • Determinar el modelo de análisis. • Realización de la Especificación de requisitos de software (ERS). • Analizar Riesgos. • Definir la calidad del producto con el cliente. • Generar el documento de análisis. • Establecimiento de un método de trazabilidad de los requerimientos.
Puntos de control	<ul style="list-style-type: none"> • Validación de requerimientos por parte del cliente.
Condiciones de aprobación	<ul style="list-style-type: none"> • Documento de análisis. • ERS.
Métodos/instrumentos	<ul style="list-style-type: none"> • Entrevistas con cliente. • Cuestionarios. • Listas de chequeo. • Modelo de análisis. • Aplicaciones de Ofimática.
Responsabilidades	<ul style="list-style-type: none"> • Jefe de proyecto: Debe aprobar el documento de análisis y validar con el analista la viabilidad de los requisitos. • Analistas: Realización de entrevistas con el cliente, desarrollo del modelo de análisis, desarrollo de la ERS. Registrar avances y resultados de cada reunión. Comunicación de los requerimientos al equipo de desarrollo y testing.
Evidencias	<ul style="list-style-type: none"> • ERS (Especificación de requisitos de software)

Diseño

Descripción	<ul style="list-style-type: none"> • Determinación del Modelo de ciclo de Vida. • Determinar la Arquitectura del sistema. • Realizar los diagramas de Clases, Actividad, Despliegue, Componentes, Estados, Secuencia y DER. • Determinación del motor de la Base de datos. • Determinar el software de desarrollo. • Determinar el software para el versionado y control de cambios de los fuentes. • Estimar el Esfuerzo que requerirá el producto. • Realizar la planificación del proyecto. • Documentación de los perfiles y habilidades requeridas.
Puntos de control	<ul style="list-style-type: none"> • En la finalización de cada incremento de cada iteración.
Condiciones de aprobación	<ul style="list-style-type: none"> • Validación contra los documentos de de diseño.
Métodos/instrumentos	<ul style="list-style-type: none"> • Técnicas de diseño. • UML. • Buenas prácticas de ingeniería de software. • Web2Project para registrar avances. • Enterprise Architect. • SVN.
Responsabilidades	<ul style="list-style-type: none"> • Jefe de proyecto: Aprobar la documentación de diseño. • Analistas: Confección de la documentación de diseño.
Evidencias	<ul style="list-style-type: none"> • Diagrama de Clases. • Diagrama de Actividad. • Diagrama de Estados. • Diagrama de Secuencia. • DER. • Diagrama de Despliegue. • Diagrama de Componentes.

Codificación

Descripción	<ul style="list-style-type: none"> • Generación de código fuente. • Generación de documentación de implementación. • Generación de pruebas unitarias. • Versionado de fuentes.
Puntos de control	<ul style="list-style-type: none"> • Finalización de cada incremento de cada iteración.
Condiciones de aprobación	<ul style="list-style-type: none"> • Pruebas de integración realizadas por el programador.

Métodos/instrumentos	<ul style="list-style-type: none"> • Eclipse 3.0 • J2EE 1.4 • SVN • Struts • JQuery • Buenas prácticas de programación. • Metodologías ágiles. • Utilización de estándares. • Utilización de frameworks. • Web2Project para registrar avances. • Mozilla Firefox. • Google Chrome
Responsabilidades	<ul style="list-style-type: none"> • Programador: Encargado de crear el código fuente. • Analista: Transmitir al programador los requisitos del usuario a través de la documentación desarrollada en la etapa de análisis.
Evidencias	<ul style="list-style-type: none"> • Fuentes subidos en el repositorio. • Estándares utilizados. • Diagrama de clases. • Diagrama de componentes. • DER.

Prueba

Descripción	<ul style="list-style-type: none"> • Realización de confección de casos de prueba. • Ejecución de casos de pruebas funcionales, unitarias, integración y de sistema. • Automatización de pruebas. • Preparación del ambiente de pruebas en lo que refiere a la preparación de los datos de la base de datos y configuración de la aplicación. • Documentación de los resultados de las pruebas.
Puntos de control	<ul style="list-style-type: none"> • Finalización de cada incremento de cada iteración. • Pruebas de integración.
Condiciones de aprobación	<ul style="list-style-type: none"> • Realización de pruebas funcionales del cliente.

Métodos/instrumentos	<ul style="list-style-type: none"> • Métodos de Caja blanca y Negra. • Normas de calidad. • Selenium para automatizar las pruebas. • Oracle VM para hacer pruebas en diferentes plataformas. • Herramientas de Ofimática para la documentación de los casos de pruebas y resultados de las pruebas. • Mozilla Firefox para realizar las pruebas funcionales. • Google Chrome para realizar las pruebas funcionales. • Web2Project para registrar avances. • JIRA para registrar las incidencias. • SVN la gestión del versionado de los documentos de la etapa de testing.
Responsabilidades	<ul style="list-style-type: none"> • Tester: Corroborar que el producto cumpla con los requisitos del cliente documentados en la fase de análisis. Realización de reporte de bugs en caso de que existieran para comunicarlo al programador y validarlo con el analista. • Programador: realización de las correcciones informadas por el tester. Comunicar al analista si existen limitación o problemas en la comprensión de los requisitos. • Analista: Validar con el cliente las funcionalidades del producto. Reportar los cambios que surjan de las reuniones.
Evidencias	<ul style="list-style-type: none"> • Documentación de casos de prueba. • Documentación de los resultados de las pruebas.

Mantenimiento

Descripción	<ul style="list-style-type: none"> • Registrar bugs solicitudes de cambios. • Realizar las correcciones informadas por los clientes. • Brindar soporte técnico online. • Realizar documentación sobre los cambios realizados. • Actualización del manual de usuario. • Realización de tareas de capacitación si fuese necesario.
Puntos de control	<ul style="list-style-type: none"> • Etapas de identificación y corrección de defectos.

	<ul style="list-style-type: none"> • Etapa de realización de pruebas unitarias.
Condiciones de aprobación	<ul style="list-style-type: none"> • Producto cumpla con las correcciones solicitadas.
Métodos/instrumentos	<ul style="list-style-type: none"> • Técnicas de revisión • Web2Project para registrar avances. • JIRA para actualizar el estado de los bugs. • Herramientas de Ofimática para documentar resultados. • SVN para mantener actualizada la documentación. • Oracle VM para reproducir las pruebas en diversos ambientes.
Responsabilidades	<ul style="list-style-type: none"> • Personal de soporte y producción: Se encarga de la distribución de la aplicación en las máquinas del cliente indicadas. Además debe documentar y corregir errores de instalación, brindar soporte al cliente y brindar capacitación. Además debe comunicar a los miembros de desarrollo los errores en producción reportados por el cliente.
Evidencias	<ul style="list-style-type: none"> • Documentación de incidencias y bugs. • Documentación de correcciones.

Proceso De Comercialización

Estudio de mercado

Descripción	<ul style="list-style-type: none"> • Analizar la competencia. • Analizar los clientes. • Realizar una estrategia, en base a los resultados de los 2 puntos anteriores de análisis. • Se obtiene un Modelo de Negocio en donde la empresa planifica las estrategias a llevar a cabo con el objetivo de generar ingresos y beneficios consiguiendo nuevos clientes. • Elaboración del Plan de Negocio, es decir la declaración formal de los objetivos de la empresa, estrategias, proyección y evaluación.
Puntos de control	<ul style="list-style-type: none"> • Modelo de Negocio • Plan de Negocio

Condiciones de aprobación	<ul style="list-style-type: none"> • Que tanto el Modelo de Negocio como el Plan de Negocio estén aprobados por el Gerente de la empresa.
Métodos/instrumentos	<ul style="list-style-type: none"> • Listado de empresas que son competencia • Listado de clientes. • Modelo de Negocio. • Plan de Negocio.
Responsabilidades	<ul style="list-style-type: none"> • Gerente de Marketing: Se ocupa de realizar todo el estudio de mercado (competencia, clientes) y elaborar estrategias de ventas. • Gerente General: Se ocupa de analizar tanto el Modelo de Negocio como el Plan de Negocio elaborado por el Gerente de Marketing. De estar de acuerdo lo aprueba, de lo contrario solicita las modificaciones correspondientes.
Evidencias	<ul style="list-style-type: none"> • Modelo de Negocio. • Plan de Negocio.

Análisis de clientes potenciales

Descripción	<ul style="list-style-type: none"> • Realizar en base al estudio de mercado, un listado con los potenciales clientes clasificándolos de acuerdo a sus niveles de necesidades de mayor a menor. • Clasificar la lista de contactos de la empresa clasificándolo de acuerdo al tipo de contacto generado previamente directo o indirecto.
Puntos de control	<ul style="list-style-type: none"> • Agenda de potenciales clientes. • Agenda de contactos.
Condiciones de aprobación	<ul style="list-style-type: none"> • Lista de potenciales clientes clasificada. • Lista de contactos clasificada.
Métodos/instrumentos	<ul style="list-style-type: none"> • Estudio de mercado. • Clasificación. • Llamados, mails, etc.
Responsabilidades	<ul style="list-style-type: none"> • Gerente de Marketing: Realiza la clasificación de los potenciales clientes y de los contactos de la empresa. • Vendedor: Recibe las listas clasificadas para comenzar a organizar el proceso de venta.
Evidencias	<ul style="list-style-type: none"> • Agenda de de potenciales clientes clasificada. • Agenda de contactos clasificada.

Llamado

Descripción	<ul style="list-style-type: none"> • Presentarse como vendedor dando a conocer su nombre, apellido e información referida a la empresa a la que pertenece. • Coordinar una reunión (fecha, hora y lugar) recordando cual fue el contacto inicial de la relación. • Envío de mail de confirmación de la reunión.
Puntos de control	<ul style="list-style-type: none"> • Registro de la llamada telefónica o de la video conferencia.
Condiciones de aprobación	<ul style="list-style-type: none"> • Validar fecha, hora y lugar de la reunión. • Verificar envío de de mail de confirmación por ambas partes.
Métodos/instrumentos	<ul style="list-style-type: none"> • Llamada telefónica. • Teléfono de línea / celular / video conferencia • Mail
Responsabilidades	<ul style="list-style-type: none"> • Vendedor: Realiza llamadas a los clientes. • Clientes: Son empresas, Pymes o particulares que pueden requerir de nuestros desarrollos para resolver sus problemas.
Evidencias	<ul style="list-style-type: none"> • Mail de confirmación de la reunión de ambas partes.

Visita

Descripción	<ul style="list-style-type: none"> • Presentación del Producto con una Demo. • Mostrar las mejoras y beneficios que se obtendrán con el Producto. • Mostrar los diferentes tipos de ayuda (Mesa de ayuda, manual, folletos, etc). • Responder las inquietudes del cliente. • De estar interesado el cliente en seguir adelante con el Proyecto, se le realizan simples preguntas para entender el objetivo y alcance de su necesidad. • Se toma nota de una primera versión de requisitos. • Coordinar un futuro contacto.
Puntos de control	<ul style="list-style-type: none"> • Planilla de reunión.
Condiciones de aprobación	<ul style="list-style-type: none"> • Tener confirmada próxima reunión con el cliente.

	<ul style="list-style-type: none"> Listado de mejoras y beneficios del Producto. Breve resumen del objetivo y alcance del Proyecto. Listado de algunos de los requerimientos.
Métodos/instrumentos	<ul style="list-style-type: none"> Oratoria. Folletos, manuales. Aplicación de Ingeniería de requerimientos. Vista rápida del sistema en algún dispositivo.
Responsabilidades	<ul style="list-style-type: none"> Gerente de Marketing: Observa, dialoga, coordina con el cliente. Vendedor: Presenta la Demo.
Evidencia	<ul style="list-style-type: none"> Planilla de reunión. Minuta de reunión.

Seguimiento

Descripción	<ul style="list-style-type: none"> Visitar al cliente en fecha, hora y lugar acordados. Realizar una presentación más avanzada del producto. Imprimir algún reporte de ejemplo de salida para que el cliente valide la calidad del producto ó algún print-screen. Responder inquietudes del cliente en cuanto al Producto. Indagar al cliente todo lo necesario para completar la versión final de la Especificación de Requisitos de Software (ERS). Obtener acuerdo pre-venta. Coordinar fecha, hora y lugar de la reunión final de venta.
Puntos de control	<ul style="list-style-type: none"> Planilla de especificación de pre venta. ERS.
Condiciones de aprobación	<ul style="list-style-type: none"> Avance del Producto. Confirmación de la reunión final de venta. ERS.
Métodos/instrumentos	<ul style="list-style-type: none"> Presentación de principales pantallas. Indagar, cuestionar, elaborar requisitos, validar. Acuerdo entre las partes.

	<ul style="list-style-type: none"> • Pre-Contrato. • ERS.
Responsabilidades	<ul style="list-style-type: none"> • Gerente de Marketing: Observa, dialoga, coordina con el cliente y firma el Pre-Contrato. • Vendedor: Muestra al cliente todo el material referido al Producto y los avances obtenidos para validar.
Evidencias	<ul style="list-style-type: none"> • Minuta de reunión. • Pre-Contrato • ERS.

Cierre de venta

Descripción	<ul style="list-style-type: none"> • Validar con el cliente la modalidad de la venta. • Clarificar todas las cláusulas del contrato con el cliente. • Se le entrega al cliente la ERS. • De estar todo validado y acordado, se completa la Factura y el Contrato. • Firma del cliente en la Factura y en el Contrato. • Se da la venta por finalizada.
Puntos de control	<ul style="list-style-type: none"> • Factura de venta. • Contrato.
Condiciones de aprobación	<ul style="list-style-type: none"> • Factura de Venta (física y digital). • Contrato firmado por el cliente.
Métodos/instrumentos	<ul style="list-style-type: none"> • Validación. • ERS. • Contrato de venta. • Factura de venta.
Responsabilidades	<ul style="list-style-type: none"> • Gerente de Marketing: Control del Contrato y la Factura. • Vendedor: Completa los datos del Contrato y la Factura.
Evidencias	<ul style="list-style-type: none"> • Contrato de venta firmado por el cliente. • Factura de venta (física y digital) firmada por el cliente.

D) Indicar en cada caso las evidencias que se requerirán en las auditorias.

Ver fila 'Evidencias' del punto c).

Usabilidad del software

El objetivo es, a partir de la Especificación de Requisitos, definir las características de usabilidad más adecuadas a ser aplicadas en el desarrollo del software, realizando las siguientes actividades:

- A) Seleccionar los atributos de usabilidad adecuados para el producto
- B) Definir las técnicas a utilizar para cada criterio
- C) Elaborar una tabla que defina las especificaciones de usabilidad con la siguiente información:

1. Atributo de usabilidad
2. Medio de medición
3. Valor a medir
4. Peor nivel aceptable
5. Mejor nivel posible
6. Nivel esperado como objetivo

- A) Seleccionar los atributos de usabilidad adecuados para el producto

Facilidad de aprendizaje: Capacidad del producto para permitir al usuario aprender su aplicación.

Eficiencia: máxima velocidad de realización de tareas del usuario.

Recuerdo en el tiempo: recuerdo del uso del sistema por parte del usuario cuando vuelve a utilizarlo tras un periodo de no utilización.

Tasa de errores: numero de errores que cometidos por el usuario mientras realiza una determinada tarea.

Satisfacción: muestra la impresión subjetiva que el usuario obtiene del sistema.

- B) Definir las técnicas a utilizar para cada criterio

<u>Facilidad de aprendizaje</u>	<u>Eficiencia</u>	<u>Recuerdo en el tiempo</u>	<u>Tasa de errores</u>	<u>Satisfacción</u>
<p><i>Especificaciones</i></p> <ul style="list-style-type: none"> • Análisis de usuarios • Análisis de tareas <p><i>Diseño</i></p> <ul style="list-style-type: none"> • Diseño de la interacción • Prototipado • Participación de los usuarios 	<p><i>Especificaciones</i></p> <ul style="list-style-type: none"> • Análisis de usuarios • Análisis de tareas <p><i>Diseño</i></p> <ul style="list-style-type: none"> • Diseño de la interacción • Prototipado • Participación de los usuarios <p><i>Evaluación</i></p> <ul style="list-style-type: none"> • Test de usabilidad 	<p><i>Especificaciones</i></p> <ul style="list-style-type: none"> • Análisis de usuarios • Análisis de tareas <p><i>Evaluación</i></p> <ul style="list-style-type: none"> • Test de usabilidad 	<p><i>Especificaciones</i></p> <ul style="list-style-type: none"> • Análisis de tareas <p><i>Diseño</i></p> <ul style="list-style-type: none"> • Prototipado • Participación de los usuarios <p><i>Evaluación</i></p> <ul style="list-style-type: none"> • Test de usabilidad 	<p><i>Especificaciones</i></p> <ul style="list-style-type: none"> • Análisis de usuarios • Análisis de tareas <p><i>Diseño</i></p> <ul style="list-style-type: none"> • Diseño de la interacción • Prototipado • Participación de los usuarios

C) Elaborar una tabla que defina las especificaciones de usabilidad

atributo de usabilidad	medio de medicion	valor a medir	peor nivel aceptable	mejor nivel posible	Nivel esperado
facilidad de aprendizaje	Puntuacion	capacidad del sistema para permitir al usuario aprender - rango de 1 a 5	1	5	4
familiaridad	Cuestionario	porcentaje de conceptos utilizados por el usuario que le resulten familiares de otros sistemas - Rango 0 a 100	50	90	70
consistencia	Puntuacion	numero de acciones equivalentes nombradas de igual forma -rango de 0 a 10	0	9	7
Potenciación de las habilidades del Usuario	Cuestionario	grado que tienen los usuarios para complementar sus habilidades con las funcionalidades del sistema - rango de 1 a 10	2	9	7

Velocidad de respuesta	Puntuacion	velocidad de respuesta ante un pedido realizado por el usuario - rango de 0 a 10	9	2	4
Tiempo para completar una tarea	Puntuacion	tiempo invertido por los usuarios para desarrollar una tarea medido de en minutos	5'	0,3'	1'
re utilización del sistema	Puntuacion	interaccion del usuario con el sistema despues de un periodo de tiempo sin usarlo - rango de 1 a 5	5	1	3
prevención de errores	Puntuacion	cantidad de Mecanismos de confirmación cuya finalidad es evitar que acciones irreversibles sean llevadas a cabo por equivocación - rango de 1 a 5	5	1	3

recuperabilidad	Puntuacion	Se ofrece al usuario la oportunidad de enmendar sus errores sin perder la información hasta ese momento procesada - rango de 1 a 5	5	1	3
Atracción	Cuestionario	Porcentaje en placer inspirado en los usuarios por la información que entrega el sistema - rango de 1 a 100	20	90	70
gratificación de las tareas	Puntuacion	Valor de satisfacción del usuario por haber usado el sistema - rango de 1 a 10	1	10	8

Solución:

La usabilidad se refiere a la facilidad con que las personas pueden utilizar una herramienta particular o cualquier otro objeto fabricado por humanos con el fin de alcanzar un objetivo concreto. El grado de usabilidad de un sistema es, por su parte, una medida empírica y relativa de la usabilidad del mismo.

Nos permiten responder a preguntas tales como:

- ¿Quiénes son los usuarios, cuáles sus conocimientos, y qué pueden aprender?
- ¿Qué quieren o necesitan hacer los usuarios?
- ¿Cuál es la formación general de los usuarios?
- ¿Pueden los usuarios realizar fácilmente sus tareas previstas? Por ejemplo, ¿pueden los usuarios realizar las tareas previstas a la velocidad esperada?
- ¿Cuánta preparación necesitan los usuarios?
- ¿Qué documentación u otro material de apoyo están disponible para ayudar al usuario? ¿Puede éste hallar las respuestas que buscan en estos medios?
- ¿Cuáles y cuántos errores cometen los usuarios cuando interactúan con el producto?
- ¿Puede el usuario recuperarse de los errores? ¿Qué han de hacer los usuarios para recuperarse de los errores?

Las respuestas a estas preguntas pueden conseguirse realizando análisis de usuarios y tareas al principio del proyecto.

Beneficios de la usabilidad:

Entre los principales beneficios se encuentran:

- Reducción de los costes de aprendizaje y esfuerzos.
- Disminución de los costes de asistencia y ayuda al usuario.
- Disminución en la tasa de errores cometidos por el usuario y del retrabajo.
- Optimización de los costes de diseño, rediseño y mantenimiento.
- Aumento de la tasa de conversión de visitantes a clientes de un sitio web.
- Aumento de la satisfacción y comodidad del usuario.
- Mejora la calidad de vida de los usuarios, ya que reduce su estrés, incrementa la satisfacción y la productividad.