

PUNTOS DE FUNCIÓN

Una mejor forma de estimar el esfuerzo requerido para desarrollar un proyecto de software es la determinación de los Puntos de Función asociados a él.

La predicción precisa del tamaño de un software ha preocupado la industria del software por más de 45 años. Los Puntos de Función se están convirtiendo en un estándar aceptado para la medición del tamaño del software.

El Análisis de Puntos de Función [Function Points] está basado en la teoría de que las funciones de una aplicación son la mejor medida del tamaño de una aplicación de software y, por lo tanto, el mejor parámetro para estimar el esfuerzo necesario para su desarrollo.

Los Puntos de Función miden el software cuantificando la funcionalidad provista al usuario en base principalmente en el diseño lógico. Para ello descompone los sistemas en componentes más pequeños de tal manera que los usuarios, desarrolladores y administradores los entiendan y analicen mejor. En el mundo de los Puntos de Función, los sistemas están divididos en cinco componentes y características generales.

El Análisis de Puntos de Función continua creciendo en popularidad, pero hay muchas discusiones y argumentos con respecto a su utilidad. La razón de estas discusiones reside en que la aplicación más común de los Puntos de Función es reportar tendencias en una organización. Es un hecho que el monitoreo de las tendencias de una organización es de mucha importancia, pero hay muchos usos de los Puntos de Función que son frecuentemente pasados por alto.

Para generar una disciplina o “práctica” de métricas de software en una empresa, es necesario el desarrollo de un esfuerzo sistemático o programa de mediciones de software. El desarrollo de un costo unitario es el primer paso que cualquier organización compleja debe tomar para entender costos y tendencias. Los Puntos de Función pueden ser la piedra angular de un programa de medición de software, pero otras medidas deben realizarse y reportarse también.

Los Puntos de Función son solo el numerador o el denominador de muchas mediciones. Un programa de desarrollo de medición de software identifica también otras medidas aparte de los Puntos de Función para recoger, reportar, monitoriar e interpretar... sus esfuerzos en desarrollo, implantación y mantención de las aplicaciones y su soporte tecnológico.

Como muchos componentes o esfuerzos en el desarrollo e implantación de software de aplicación es subcontratado o externalizado por la empresa, es necesario también realizar un “Análisis de Contratos de Software”, estableciendo las condiciones mínimas de información sobre métodos de trabajo, experiencia de participantes, subdivisión de tareas, establecimiento de productos o entregables y registro de

esfuerzos tiempos y condiciones en las que el desarrollo “efectivamente” se llevó a cabo.

Además, el establecimiento de garantías y acuerdos en los contratos de software puede ayudar a ambas partes, tanto el vendedor como el comprador de software a lograr los objetivos del proyecto de software. Aún más importante, las garantías y los acuerdos pueden ayudar a reducir el costo de un conflicto costoso. Se deben establecer a lo menos los siguientes acuerdos y garantías:

1. Compatibilidad con el software existente y la interfase de los sistemas
2. Conformidad del sistema de software instalado con los requerimientos originales
3. Establecimiento y conformidad de los criterios de ejecución y otras métricas.
4. Información sobre el avance y cierre del proyecto que será entregada, incluyendo las métricas acordadas y registradas.
5. Costo de mejoramiento (o cambio) de los requisitos o especificaciones originales
6. Costo de actividades de mantención junto con una definición detallada de mantención
7. Políticas de cancelación de contratos
8. Definición de términos computacionales usados en los contratos
9. Políticas de arbitraje en el caso de que cualquiera de las garantías anteriores se pusieran en disputa (“.. un mal acuerdo es mejor que un buen juicio..”)

Los Puntos de Función se caracterizan por:

1. Ser un método independiente de las herramientas de análisis, diseño y programación, debido a que se preocupa sólo de la complejidad de las funciones a implementar.
2. Requerir de una descomposición funcional del proyecto de software a realizar, en términos tales que se detecten todas las piezas elementales que componen el producto final. (Funciones elementales).
3. Estimar la "cantidad de Puntos de Función" de las funciones medidas, se realiza contando la cantidad de entradas, salidas, archivos, consultas e interfaces que utiliza. A mayor cantidad, mayor es el "peso de complejidad" que se le asignará.
4. Ajustar la estimación del esfuerzo requerido, por la vía de determinar la presencia de ciertos elementos que dificultan el desarrollo del proyecto.
5. Permitir realizar una estimación del esfuerzo requerido (y por ende de la duración del proyecto) en etapas tempranas del proyecto (Etapa de Análisis).

Los beneficios de utilizar esta técnica son evidentes, permite realizar la estimación del trabajo requerido en una etapa temprana del proyecto y es independiente del entorno tecnológico a utilizar.

Cinco Componentes Principales

Una buena fuente de información para determinar las entradas externas (EI's) son las formas de pantalla y las formas de diálogo, o bien cualquier otra forma de entrada. Las entradas adicionales de otras aplicaciones se deben inventariar aquí. Las entradas de otras aplicaciones deben actualizar los archivos lógicos internos (ILF's) de la aplicación que se está midiendo.

External Inputs (EI) - La entrada externa es un proceso elemental en el cual los datos cruzan los límites de afuera hacia adentro. Estos datos pueden venir de una pantalla de entrada de datos o de otra aplicación. Los datos son usados para mantener uno o más archivos lógicos internos (ILF's). Los datos pueden ser de información de control o del negocio. Si los datos son de información de control, no se tiene que actualizar un archivo lógico interno.

External Outputs (EO) - Una salida externa es un proceso elemental en el cual los datos derivados cruzan los límites de adentro hacia afuera. Los datos crean reportes o archivos de salida enviados a otras aplicaciones. Estos reportes y archivos son creados de uno o más archivos lógicos internos (ILF's) o archivos de interfase externos (EIF's).

Los Datos Derivados son datos que son procesados más allá de la edición directa de información de archivos lógicos internos. Los datos derivados son usualmente el resultado de algoritmos o cálculos. Los datos derivados ocurren cuando uno o más elementos de datos son combinados con una fórmula para generar o derivar elementos de datos adicionales.

External Inquiry (EQ) - Una consulta externa es un proceso elemental con componentes de entrada y de salida que resultan en la adquisición de datos de uno o más archivos lógicos internos (ILF's) o archivos de interfase externos (EIF's). El proceso de entrada no actualiza ningún archivo lógico interno (ILF), y el proceso de salida no contiene datos derivados.

Internal Logical Files (ILF) - Un archivo lógico interno es un grupo de datos definidos por el usuario que están relacionados lógicamente, que residen en su totalidad dentro de los límites de la aplicación y que son mantenidos a través de entradas externas (EI's).

External Interface Files (EIF) - Un archivo de interfase externo es un grupo de datos definidos por el usuario que están relacionados lógicamente y que solo son usados para propósitos de referencia. Los datos residen enteramente fuera de la aplicación y son mantenidos por otra aplicación. El archivo de interfase externo es un archivo lógico interno (ILF) para otra aplicación.

El Factor de Ajuste

El Factor de Ajuste (FAV) está basado en características generales del sistema [General System Characteristics] (GSCs) que evalúan la funcionalidad general de la aplicación que se está midiendo. Cada característica tiene asociada una descripción que ayuda a determinar su nivel de influencia en una escala de cero (sin influencia) a cinco (influencia fuerte). El manual IFPUG Counting Practices Manual (CPM 4.0) proporciona criterios de evaluación detallados para cada GSC; la siguiente tabla provee un resumen de cada una.

Características Generales del Sistema y su descripción

1. Comunicación de datos
¿Cuántas herramientas de comunicación hay para ayudar en la transferencia o intercambio de información de la aplicación o sistema?
2. Procesamiento de datos distribuidos
¿Cómo son manejados los datos distribuidos y las funciones de procesamiento?
3. Nivel de ejecución
¿El tiempo de respuesta o el nivel de eficiencia es requerido por el usuario?
4. Configuración más usada
¿Qué tanto se usa la plataforma de hardware en donde la aplicación se va a ejecutar?
5. Nivel de transacciones
¿Qué tan frecuentemente se ejecutan las transacciones al día, semana, mes etc.?
6. Captura de datos En Línea
¿Qué porcentaje de información se captura En Línea?
7. Eficiencia del usuario final
¿Se diseñó la aplicación pensando en la eficiencia del usuario final?
8. Actualización En Línea
¿Cuántos ILF's se actualizan en transacciones En Línea?
9. Procesamiento complejo ¿La aplicación tiene mucho procesamiento lógico o matemático?
10. Reusabilidad
¿La aplicación se desarrolló para cumplir una o muchas necesidades del usuario?
11. Facilidad de Instalación
¿Qué tan difíciles son la conversión y la instalación?

12. Facilidad de Operación

¿Qué tan efectivos y/o automatizados son los procedimientos de inicio, respaldo y recuperación?

13. Múltiples Sitios

¿La aplicación se diseñó, desarrolló y soportó específicamente para ser instalada en múltiples sitios para varias organizaciones?

14. Facilidad de mantención

¿La aplicación se diseñó, desarrolló y soportó específicamente para facilitar la mantención?

Una vez que el VAF y cada componente han sido identificados, se acomodan y cuentan usando un conjunto de cálculos predeterminados. Para los lectores interesados en aprender la mecánica del conteo de Puntos de Función, se recomienda contactar al grupo IFPUG y pedir el manual: Counting Practices Manual Version 4.0.

El uso de Puntos de Función para ayudar a calcular el costo real del software.

La mayoría de las organizaciones subestima en gran medida el costo del software. El costo real del software es la suma de todos los costos durante la vida de un proyecto, incluyendo los mejoramientos esperados y los costos de mantención. De hecho, el cálculo real debería ser el valor presente de todos los desarrollos, mejoras, y costos de mantención esperados durante la vida del proyecto. Este tipo de análisis demuestra la recompensa de invertir en un diseño y análisis de primera. Entre más se invierta en un buen diseño, se va a ahorrar más en futuros costos de mantención y mejoras. Es importante tener un costo unitario para evaluar la inversión inicial y comparar ésta con los gastos posteriores. El costo unitario puede ser horas/PF o \$/PF. Los incrementos en la inversión inicial deben reducir el costo unitario de actividades de mejora y mantención futuras.

El uso de Puntos de Función para ayudar a estimar el costo de proyectos, la programación y el esfuerzo.

La estimación exitosa usando Puntos de Función se basa en varias técnicas: Top-Down, Analogía y Consejo de Expertos. La estimación Top-Down es una técnica de estimación que calcula el programa entero, costo y esfuerzo usando parámetros amplios. Los parámetros amplios y las comparaciones están basados en datos históricos usando técnicas estimativas de Analogía. El Consejo de Expertos se obtiene de expertos con experiencia en proyectos similares o experiencia en el uso de Puntos de Función.

La comparación de proyectos con otros similares es una actividad crítica para lograr una estimación exitosa. Cuando se evalúan proyectos similares, se debe considerar lo siguiente:

- Tipo de plataforma de hardware - Mainframe, Cliente-Servidor, PC, Internet
- Tipo de lenguaje – Adabas, Informix, Visual Basic, C, C++, Java o PHP
- Tipo de proyecto - Software del Sistema, Software intermedio, Software de aplicación
- Tipo de sistema operativo: MVS, Windows, Linux, Unix

Una vez que los proyectos han sido determinados, obtener los siguientes datos:

- Medida histórica de entrega (horas por Punto de Función) de proyectos similares
- Programas históricos (duración de programas por Punto de Función) de proyectos similares
- Costos históricos (costo en \$, U.F. o US\$ por Punto de Función)

Una vez que el tamaño del proyecto se ha determinado en Puntos de Función, se puede calcular la estimación de horas, costo y cantidad de programas. Los cálculos se deben hacer con datos de proyectos similares como se describió anteriormente.

Por ejemplo, si se determina que el tamaño del proyecto actual es de 500 Puntos de Función y la medida de entrega de un proyecto similar es UF.2 por Punto de Funcionalidad, entonces el costo total esperado para el proyecto sería $UF.2 \times (UF/Punto\ de\ Función) \times 500\ PF's = UF.1,000$. Cálculos similares pueden hacerse para programas, duración y horas.

El uso de Puntos de Función para ayudar con las negociaciones de contrato.

Los administradores de contratos pueden usar su conocimiento en Puntos de Función para construir y manejar proyectos basados en el precio por Punto de Funcionalidad y también en la comparación de los precios de los vendedores. Estas personas establecen un uso efectivo en cuanto a costo, de terceras partes, en el desarrollo, validan las propuestas basados en el tamaño de Puntos de Función y pueden evaluar el impacto de proyectos cancelados.

Los Puntos de Función pueden ser usados para ayudar a especificar los productos claves a entregar a un vendedor, para asegurar que los niveles apropiados de funcionalidad van a ser entregados y desarrollar medidas objetivas de efectividad de costos y calidad. Son los más efectivamente usados en contratos de precio fijo como un medio para especificar exactamente lo que se va a entregar.

Desde una perspectiva interna, el manejo exitoso de los contratos de precio fijo depende absolutamente en la representación precisa del esfuerzo. La estimación de este esfuerzo (en el ciclo de vida completo) puede ocurrir solo cuando una métrica normalizada, tal como la proveída por los Puntos de Función, se aplica.

Resumiendo, el análisis de Puntos de Función provee el mejor método objetivo para medir los proyectos de software, y para manejar el tamaño de los proyectos de software durante su desarrollo.

PF es el mejor método de manejar el riesgo en dos vertientes. Primero, el cliente (usuario/especificador) puede aceptar más fácilmente el riesgo para un determinado tamaño de proyecto de software (en Puntos de Función). Segundo, el desarrollador puede más fácilmente aceptar los riesgos para el costo de producción (el costo por Punto de Funcionalidad). Adherirse a un conteo consistente de Puntos de Función optimiza esta relación y facilita el desarrollo en línea y bajo presupuesto.

La asignación de precios de "software externo" (p.ej. el diseñado para uso fuera de la organización) puede ser encauzado directamente al esfuerzo de producción cuando se requieren métricas funcionales. Si un desarrollador de software sabe exactamente cuál va a ser su costo interno de desarrollo de un Punto de Funcionalidad, se puede incorporar a los algoritmos de costeo usados para determinar los precios externos. Sin un entendimiento claro del tiempo y esfuerzo por Punto de Funcionalidad, la asignación de precios a los paquetes de software continuará siendo difícil.

El uso de Puntos de Función para desarrollar un estándar de establecimiento de métricas

Por supuesto, los Puntos de Función necesitan usarse en asociación con las otras medidas. De hecho, los Puntos de Función por sí mismos proveen poco o nada de beneficio. Muchas métricas necesitan ser reportadas al nivel organizacional. Por ejemplo, tanto la métrica de productividad/costo como la métrica de calidad necesitan ser reportadas al nivel organizacional. Las métricas de productividad/costo son usadas para demostrar la medida y el costo de la funcionalidad que se está entregando. Las métricas de calidad son usadas para demostrar niveles existentes de calidad y para monitoriar los esfuerzos continuos de mejoramiento en el proceso de desarrollo del software. Estas métricas deben ser monitoriadas y estudiadas en sus tendencias.

Productividad / Métricas de Costo

Costo por Punto de Funcionalidad: mide el costo promedio para entregar o mantener un Punto de Funcionalidad. Puntos de Función por Staff por Mes: mide el número promedio de Puntos de Función entregados por mes de esfuerzo aplicado

Métricas de Calidad

1. Defectos por Puntos de Función Instalados: correlaciona la calidad del software al tamaño de la aplicación. Horas de Mantenimiento por Puntos de Función Instalados: correlaciona los esfuerzos de soporte al tamaño de la aplicación para el software instalado actualmente y los sistemas nuevos. Las aplicaciones con altas proporciones son buenas candidatas para reingeniería o para reemplazo. Comparar la historia con lo reciente permite asegurar que las mejoras en el proceso de entrega sean efectivas. Las métricas deben proveer suficiente granularidad para mostrar tendencias generales, identificar áreas problemáticas, y demostrar el progreso. El tratar de lograr métricas perfectas retarda y causa que gaste más en medición que en acción.

Para utilizar este método, se deben seguir los siguientes pasos:

1. Descomponer la aplicación a construir, en funciones elementales a implementar. Para esto se puede utilizar las técnicas de Descomposición Funcional, Diagramas de Flujos de Datos, o simplemente generar un listado de las funciones a contemplar.
2. Para cada función, se debe cuantificar los siguientes parámetros:

PARAMETRO	Q	Factor de peso			Subtotal (1)
		Simple	Medio	Complejo	
Número de Entradas	_____	x 3	4	6	_____
Número de Salidas	_____	x 4	5	7	_____
Número de Consultas	_____	x 3	4	6	_____
Número de Archivos	_____	x 7	10	15	_____
Número de Interfaces	_____	x 5	7	10	_____
		Cuenta (2)			

Donde:

- (1) Se obtiene como: Q x factor de peso escogido.
 (2) Se obtiene como la suma de las cantidades de la columna 'Subtotal'.
3. Obtener el total de puntos de función para la aplicación completa:

$$\text{Cuenta-Total} = \text{SUMA} (\text{Cuenta})$$

4. Calcular los Puntos de Funciones totales, utilizando la relación:

$$\text{PF} = \text{Cuenta-Total} * [0,65 + 0,01 * \text{SUMA}(\text{Fi})]$$

Donde, Fi (i=1 hasta 14) se obtiene ponderando cada factor contenido en la Tabla N° 1 (en la página siguiente).

5. Determinar el estándar de productividad (PF/h-m) que utilizará la organización, medido en Puntos de Función por hombre-mes. Es decir, cuántos PF/mes en promedio, producirá un integrante del equipo humano. [Algunas organizaciones utilizan valores entre 50 y 75 PF/h-m].

6. Finalmente, calcular el esfuerzo y duración del proyecto utilizando las siguientes ecuaciones:

$E = \frac{PF}{PF/h-m} \text{ [hombre-mes]}$
$D = \frac{E}{N} \text{ [meses]}$

Donde, **N**: N° de personas disponibles. **D**: Distribución estimada del proyecto

ASPECTO	DESCRIPCION
1. Comunicación de datos	¿ Se requiere implementar mecanismos de comunicación de datos ?
2. Procesamiento distribuido	¿ Existen funciones que requieran de procesamiento distribuido ?
3. Nivel de desempeño	¿ Es crítico el desempeño del sistema para el éxito de la gestión ?
4. Disponibilidad del software	¿ El sistema será ejecutado en un ambiente operativo existente y fuertemente utilizado ?
5. Volúmen de transacciones	¿ Es grande el número de transacciones que el sistema deberá soportar ?
6. Ingreso interactivo	¿ Requiere el sistema un alto y sofisticado nivel de ingreso interactivo de datos al sistema ?
7. Interfaz de usuario	¿ Es muy compleja y variada la interfaz hacia el usuario (múltiples pantallas, help línea, amistosidad, etc) ?
8. Actualización en línea	¿ Se actualiza la B.D. en línea, esto es, a partir de actualizaciones interactivas ?
9. Complejidad interna	¿ Existen un alto nivel de programación de reglas de excepción, cálculos complejos, etc. ?
10. Reusabilidad	¿ Se ha de diseñar el software para ser re-utilizado en otros proyectos ?
11. Facilidad de instalación	¿ Están incluidas en el diseño de la solución la conversión de datos y la implementación ?
12. Complejidad externa de procesamiento	¿ Qué tan complejas son las entradas, salidas y consultas del sistema ?
13. Multiplicidad	¿ El sistema deberá soportar múltiples instalaciones para diferentes organizaciones (o sucursales) ?
14. Adaptabilidad	¿ Se diseñó la solución para ser fácilmente modificable y mantenible ?

Tabla 1. Factor de Ajuste del Esfuerzo (FAE)

FACTOR	SIGNIFICADO
0	No presente
1	Incidental
2	Moderado
3	Medio
4	Significativo
5	Esencial

Tabla 2. Valores para los FAE

Taller: Aplicación de Puntos de Función (P. Aguilera)

Con el fin de aplicar el método denominado "Puntos de Función", se enuncian a continuación los principales requerimientos asociados a un pequeño sistema.

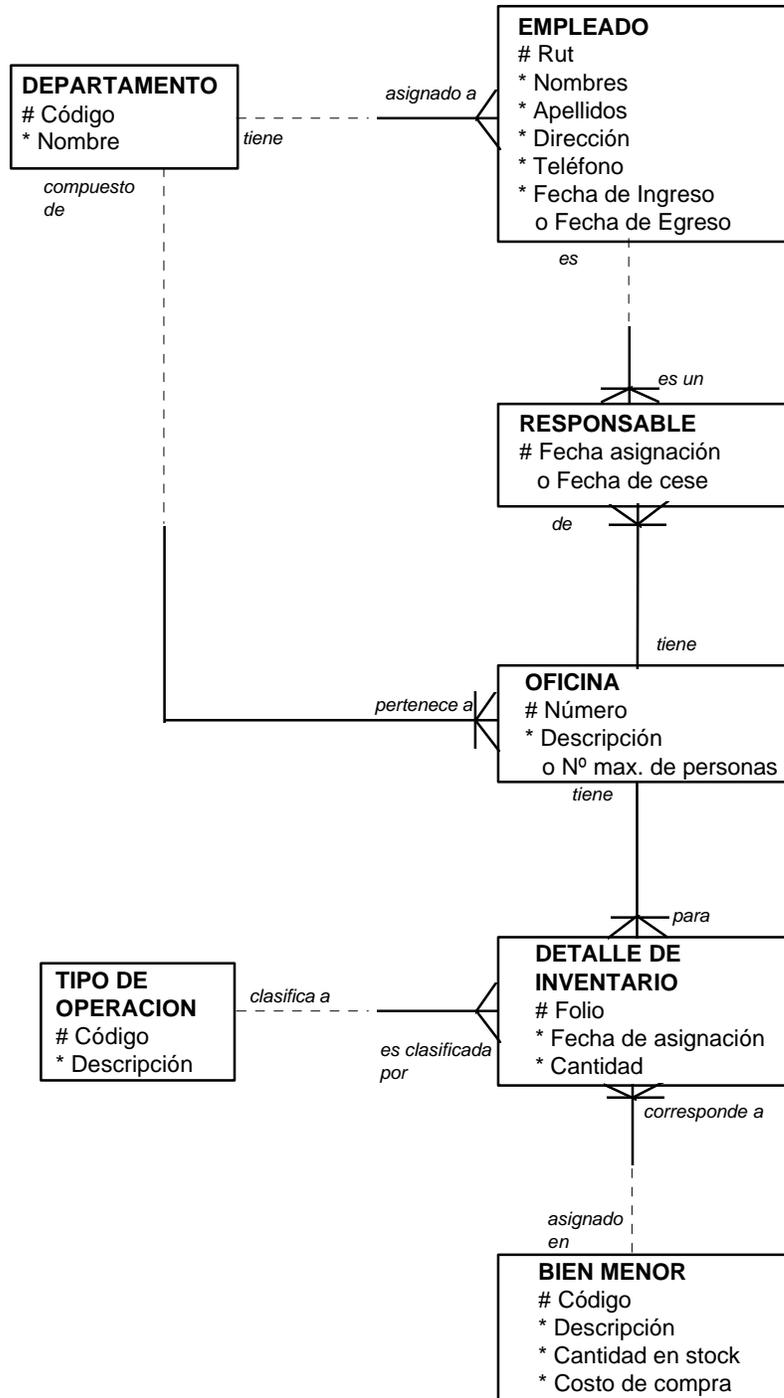
Se desea controlar el denominado Inventario de Bienes Menores de una empresa, también conocido como "INVENTARIO MURAL" (debido a que se trata de los Bienes que están en las Oficinas y se pega el informe de los bienes en las paredes de éstas).

De la recopilación de antecedentes de las entrevistas sostenidas con quienes administran en este sistema, se pudo detectar la siguiente información y requerimientos:

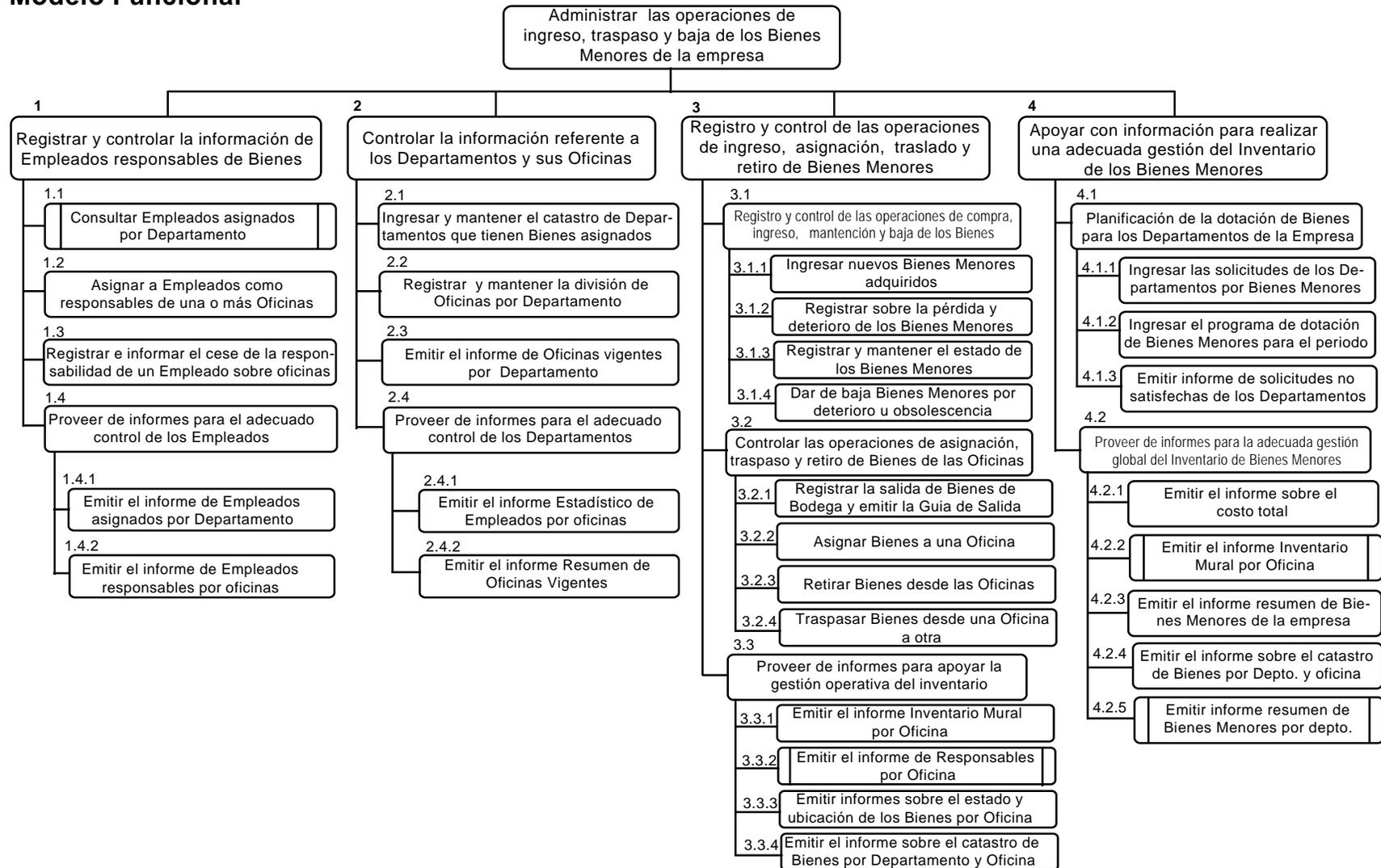
- 1 Los Empleados de la empresa se asignan a los Departamentos en los que trabajan (Departamento en el sentido de Unidad Organizacional)
- 2 Cada Departamento se compone de Oficinas (físicas), en las que habitan las personas del Departamento. La Oficina se identifica con un número correlativo por cada Departamento.
- 3 La asignación de Bienes (Inventario) es a las Oficinas, es decir los Bienes Menores son inventariados como pertenecientes a la Oficina en la que están asignados.
- 4 Cada Oficina tiene uno (o más) responsable(s) por los Bienes que están en ella. A su vez, una persona de un Departamento, puede ser responsable de varias Oficinas (P. ej. la secretaria).
- 5 Se puede concluir que una persona es responsable por las Oficinas (no por los Bienes individuales), y debido a esto, lo es de todos los Bienes que están contenidos en 'sus' Oficinas.
- 6 Se sabe que para efectos del sistema un bien es genérico. Un bien es genérico cuando existe un solo código para agrupar a un conjunto de Bienes (por ejemplo, un solo código para las 'Sillas de cuero negro con 5 ruedas'). Un Bien es específico, si existe un código distinto (único) para cada Bien (en el caso de las sillas, éstas tendrían códigos distintos).
- 7 Agregue la funcionalidad básica para permitir un adecuado nivel de gestión (esto es, las consultas e informes mínimos para administrar el sistema).

En las dos próximas páginas se encuentran los modelos de funciones y datos necesarios para aplicar el método. Utilice el formulario que permite, en forma resumida, aplicar el método y tabular la información.

Modelo de Información



Modelo Funcional



Conclusiones

A la luz de los resultados y de los entornos tecnológicos actuales, en los que se está utilizando cada vez más herramientas 4GL, SABDR y CASE; todas las cuales ahorran una cantidad sustancial de esfuerzo en el proceso de Análisis, Diseño, Construcción y Documentación de los productos de software; se intuye que los únicos métodos válidos y factibles de utilizar son Puntos de Función y estándar de Consultoría.

Para que este eficaz y simple método tenga éxito, es preciso que las organizaciones realicen un proceso formal de desarrollo de sistemas. Esto es, que utilicen una metodología en forma rigurosa, de manera que en la Etapa de Análisis se obtenga un *Modelo Funcional* del (los) sistema (s) a construir, que permita medir las funciones elementales detectadas con un alto grado de certeza.

Para obtener este Modelo Funcional, se puede recurrir a alguna de las siguientes técnicas:

TECNICA	RESULTADO	ETAPA
Modelamiento de Procesos de Negocios	Modelo de Procesos de los sistemas a construir. Al final del proceso, quedarán documentadas las funciones elementales a automatizar	Estrategia
Descomposición Funcional	Una Jerarquía de Funciones, en cuyas 'hojas' están contenidas las funciones elementales a implementar.	Análisis
Diagramas de Flujos de Datos	Se trata de utilizar la técnica de modelamiento de funciones, propiciada por la metodología del Análisis Estructurado. En esta técnica, las funciones elementales se encuentran en el último plano de la descomposición	Análisis
Listado informal	Simplemente se realiza un check-list de las funciones del sistema que deberán ser implementadas. En este caso no existe un enfoque sistémico para el análisis del sistema, sino más bien se trata de un rápido proceso de recuento de funciones (quizás con el único propósito de estimar esfuerzo mediante la técnica de estándar de consultoría).	Estrategia o Análisis