

UNIDAD 4

Ingeniería de Software

Estimaciones de Software



Conceptos de Estimaciones de Software

Fundamentos de Estimaciones

¿ Qué preguntas nos hacemos al estimar ?

- *La mayoría de las veces nos preguntamos:*
 - ¿ Cuánto **tiempo** vamos a tardar en desarrollar el proyecto ?
 - ¿ Cuantas **hs/pers** se necesitan para desarrollarlo ?
 - ¿ Cuánto **va a costar** el proyecto ?

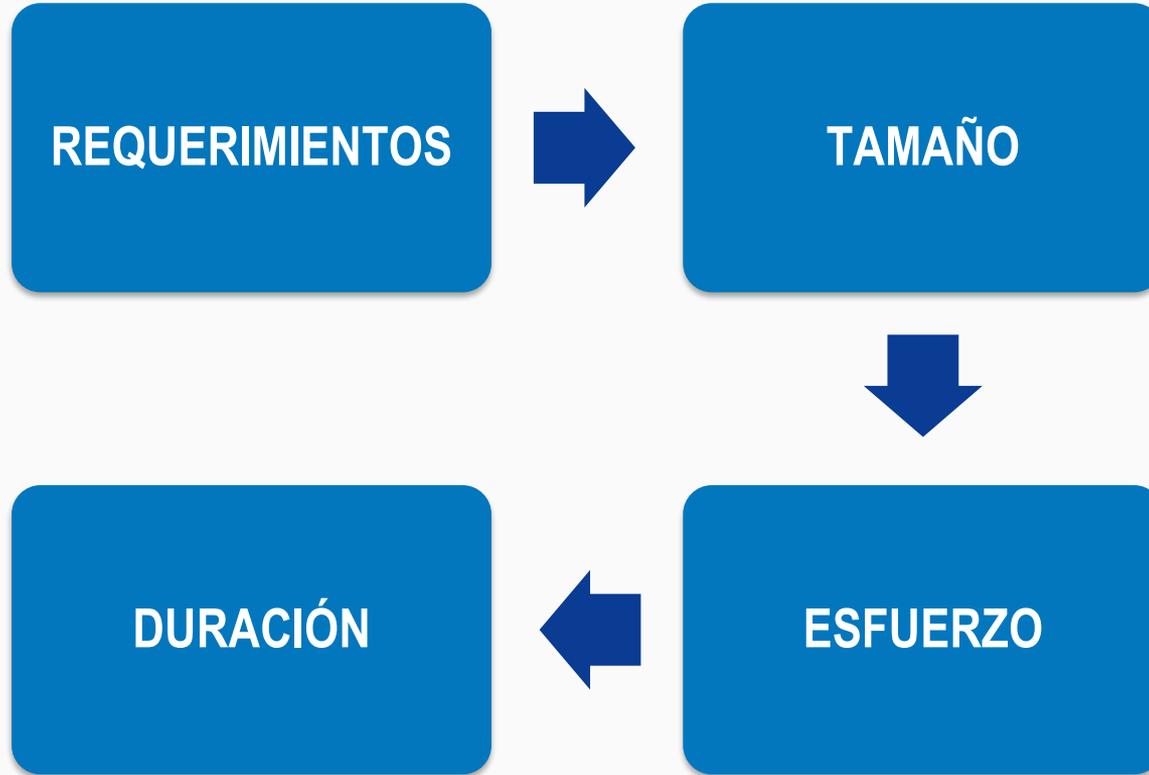
Son pocas las veces que nos preguntamos:

- *¿Cuál es el tamaño de lo que tenemos que construir ?*
 - Esta pregunta debería ser la primera para poder pasar a las anteriores !!!
 - La realidad parece indicarnos que el tamaño se calcula informalmente y solo formalizamos el esfuerzo, ventana y costo
 - Debemos ser lo mas rigurosos posibles con la estimación del tamaño



Fundamentos de Estimaciones

EL PROCESO DE ESTIMAR



Fundamentos de Estimaciones

¿Por qué fallan las estimaciones?

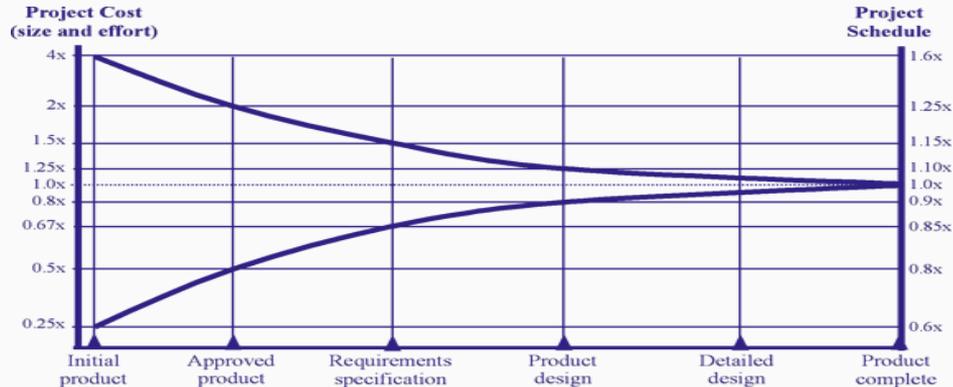
- *Optimismo*
- *Estimaciones informales (“estomacales”)*
- *No hay historia*
- *Mala definición del alcance*
- *Novedad / Falta de Experiencia*
- *Complejidad del problema a resolver*
- *No se estima el tamaño*
- *Porque la estimación fue buena pero cuando empieza el proyecto:*
 - *Mala administración de los requerimientos*
 - *No hay seguimiento y control*
 - *Se confunde progreso con esfuerzo*



Fundamentos de Estimaciones

“Nivel de incertidumbre”

- El “cono de la incertidumbre” define niveles estadísticos predecibles de la incertidumbre de las estimaciones en cada etapa del proyecto
- Cuanto más refinada la definición, mas exacta será la estimación



Fundamentos de Estimaciones

Tips para estimar ...

- *Diferenciar entre estimaciones, objetivos y compromisos*
- *Asociar a las estimaciones un % de confiabilidad*
- *Es recomendable presentar las estimaciones como rangos en lugar de un único valor*
- *Siempre presentar junto con la estimación, los supuestos que se tuvieron en cuenta para llegar a la misma*



Fundamentos de Estimaciones

Tips para estimar ... (cont.)

- *Tener presente la **Ley de Parkinson**: “Toda tarea se expande hasta ocupar el tiempo que tiene asignado”*
- *Considerar todas las actividades relacionadas al desarrollo de sw, no solamente codificación y testing (análisis, diseño, actividades de SCM, testing, etc.)*
- *No asumir que solo por el paso del tiempo y de las fases de un proyecto se avanza con menor incertidumbre en las estimaciones (Cono de la incertidumbre)*
- *Recolectar datos históricos para tener como referencia*

Fundamentos de Estimaciones

Estimaciones creíbles ...

- *Las estimaciones las hacen las personas, no herramientas ni modelos.*
 - Se necesitan juicios razonables y compromisos con los objetivos organizacionales que no se pueden delegar a modelos automáticos
- *Las estimaciones se basan en comparaciones.*
 - Cuando las personas estiman buscan similitudes y diferencias con proyectos previos
- *Para que la gente pueda estimar necesitamos **historia** de proyectos pasados para poder comparar*
 - Las estimaciones se pueden mejorar colectivamente juntando historia
- *Un método creíble debe ser de caja blanca*

Fundamentos de Estimaciones

Ciclo de Estimación

- **Estimar:** Comenzar x estimar el tamaño para derivar el esfuerzo y el costo
- **Medir:** Mientras evoluciona el proyecto, medir el tamaño, el esfuerzo y costo incurrido
- **Registrar:** Dejar claras las mediciones tomadas
- **Analizar:** Razones de desvíos, supuestos que quizás variaron, temas no contemplados, etc...
- **Calibrar:** Ajustar c/u de las variables y parámetros que intervienen en el proceso de estimación
- **Volver a estimar:**
 - El mismo proyecto pero ahora con mas información que al comienzo del mismo
 - Nuevos proyectos con el proceso ajustado por la "calibración"





Métodos de Estimación

Métodos de Estimación

- **Métodos rudimentarios**

- *Juicio Experto*
- *Pert*
- *Wideband Delphi*
- *Planning Poker*

- **Métodos paramétricos**

- *Function Points*
- *Use Case Points*
- *Story Points*
- *Objects Points*



Métodos de estimación

Recomendaciones - ¿Qué hacer?

- *Juntar historia de todos los proyectos*
 - *Registrar todas las lecciones aprendidas*
 - *Tener en cuenta la importancia de la comunicación*
- *Desarrollar un método de estimación adecuado a la instalación*
 - *Basarse en los conocidos y crear el propio*
 - *Lo importante es su eficacia*
- *Apoyarnos en herramientas*
 - *Permiten ayudarnos en la implementación del “ciclo dorado” para implementar un método*



Métodos de estimación

Recomendaciones - ¿Qué NO hacer?

- *Descartar los métodos (Rechazarlos porque los consideramos inaplicables)*
 - *Rechazarlos porque los consideramos inaplicables*
- *Utilizar métodos elaborados sin experiencia o información suficiente*
 - *Los métodos elaborados requieren de habilidades que hay que desarrollar*
- *Ignorar los supuestos*
 - *Siempre deben estar claro al comienzo, dejando claro el nivel de certidumbre de la estimación.*





Preguntas



Apéndices

OBJECT POINTS

- *Los “objetos” contemplan pantallas, reportes & módulos*
 - No está relacionado necesariamente con objetos de OOP
 - *El método original contempla solo estos tres tipos de objetos*
 - Implementaciones ad-hoc del método consideran otros tipos de componentes como (stored procedures, clases, scripts SQL, etc ...)
- *Se asigna a cada componente un peso de acuerdo a su clasificación por complejidad*
- *Considera como factor de ajuste el porcentaje de reuso de código*

Métodos Paramétricos

OBJECT POINTS (cont.)

- *Cada objeto es clasificado de acuerdo a su nivel de complejidad en:*
 - Simple
 - Medio
 - Difícil



	Number and sources of data tables		
Number of Views Contained	Total < 4	Total < 8	Total 8+
<3	simple	simple	medium
3-7	simple	medium	difficult
8+	medium	difficult	difficult

	Number and source of data tables		
Number of Sections Contained	Total < 4	Total < 8	Total 8+
0-1	simple	simple	medium
2-3	simple	medium	difficult
4+	medium	difficult	difficult

Métodos Paramétricos

OBJECT POINTS (cont.)

- *Luego se le brinda un peso a cada nivel de complejidad (teniendo directa proporción con el esfuerzo que requiere la implementación de c/u)*

	Weight		
Type	Simple	Medium	Difficult
Screen	1	2	3
Report	2	5	7
Modules	10	10	10

- *Finalmente sumando la cantidad de objetos de cada complejidad con sus respectivos pesos nos da como resultado los **OPs***
- *Teniendo en cuenta el reuso \rightarrow **NOP (NewOP)** = $OP * [(100 - \% \text{ Reuso}) / 100]$*