

## Estimación con COSMIC

**Requerimiento Funcional :** Creación de Páginas de Sitios

### Puntos de Función del RF según COSMIC

- *Entrada* : El usuario solicita la creación de una página.
- *Lectura* : El sistema revisa si el usuario está registrado como usuario en la Database.
- *Entrada* : El usuario ingresa todos los datos del sitio que desea agregar.
- *Escritura* : El sistema introduce los datos del sitio del usuario dentro de la Database.
- *Salida* : El sistema devuelve al usuario su página creada.

### Estimación CPE (Costo, Plazo, Esfuerzo)

# Entradas	# Salidas	# Lecturas	# Escrituras	Functional Size (CFP)
2	1	1	1	5

El Tamaño Funcional del Proyecto equivale a **5 CFP** (COSMIC Function Points).

Si consideramos una productividad promedio de **3 CFP/persona-mes** y considerando un coste de mantenimiento mensual de **30,000 pesos mensuales** (Considerando un pago mensual para cada miembro de **6,000 pesos**) entonces cada punto de función cuesta:

$$(\$30,000)/(3 \text{ CFP}) = \$10,000/\text{CFP}$$

Entonces el coste total del proyecto es:

$$(5 \text{ CFP}) * (\$10,000/\text{CFP}) = \$50,000$$

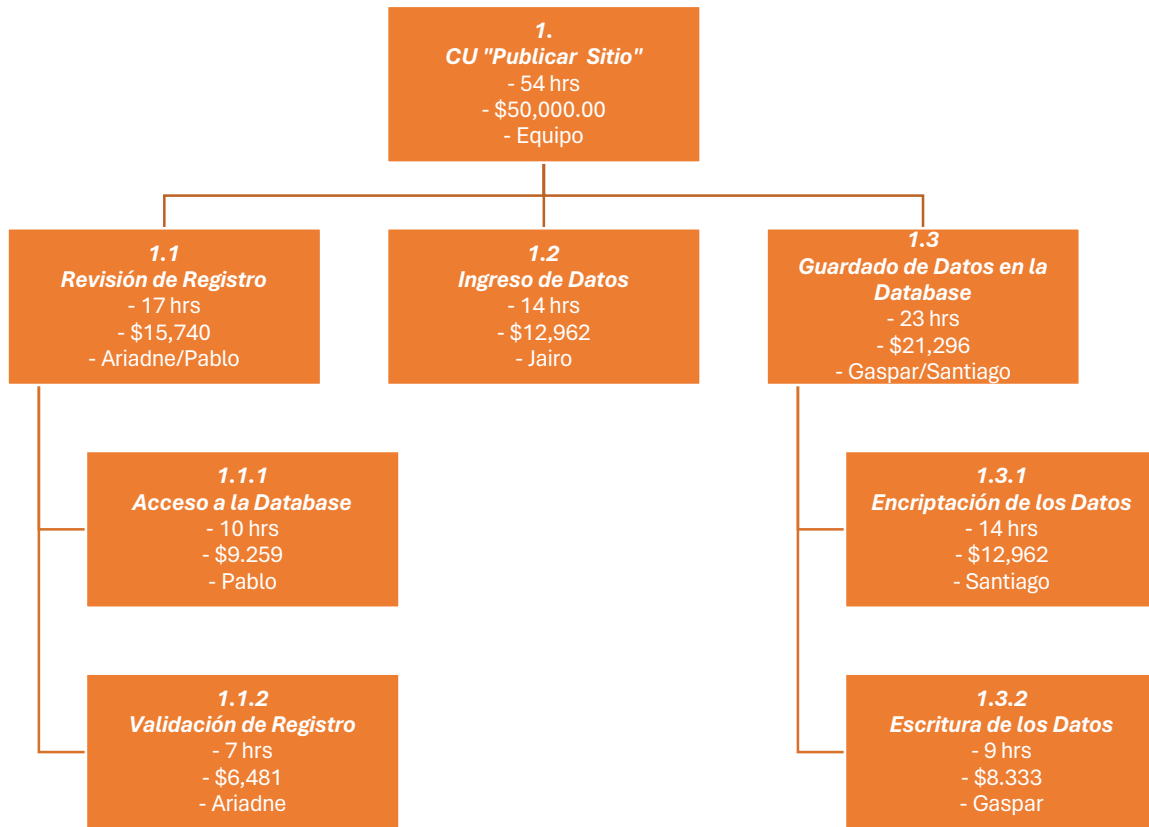
Por su parte, el esfuerzo requerido se calcularía como:

$$(5\text{CFP}) / (3 \text{ CFP/persona-mes}) = 1.66 \text{ persona-mes}$$

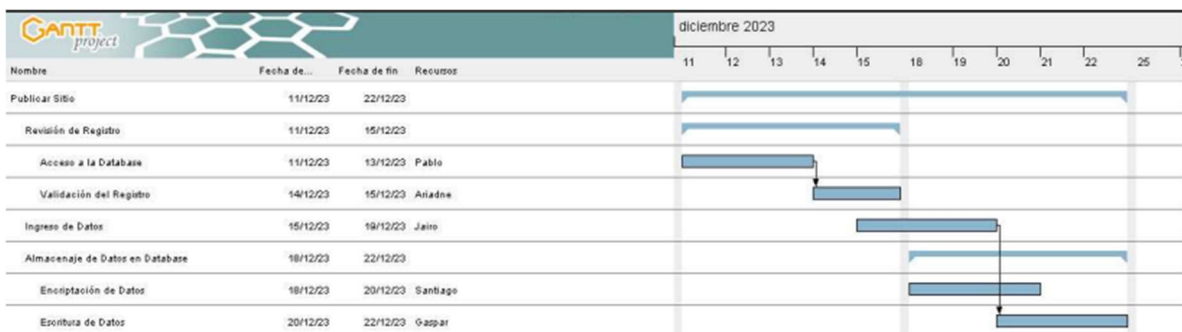
Considerando un equipo de desarrollo de 5 personas entonces el tiempo necesario para terminar los puntos de Función sería:

$$1.66 \text{ persona-mes} / 5 \text{ personas} = 0.33 \text{ meses}$$

## WBS



## Diagrama de Gantt



## Comparación de Modelos de Calidad McCall/Boehm

Los modelos de calidad propuestos por McCall y Boehm son dos modelos diferentes para la evaluación de la calidad del software. Ambos modelos se basan en un conjunto de factores que afectan la calidad del software. Algunas de las características principales de cada uno son:

McCall	Boehm
Se centra en la calidad del Producto	Se centra en la calidad del proceso
Establece 3 perspectivas para identificarlo: Operación, Revisión y Transición	Se divide en una jerarquía de 2 niveles (Bajo y Alto) En el alto hay 3 características: Utilidad, Capacidad de Mantenimiento y Movilidad
Cada Perspectiva tiene varios factores (11): Operación : Corrección, Fiabilidad, Eficiencia, Integridad, Usabilidad Revisión : Facilidad de Mantenimiento, Facilidad de Evaluación, Flexibilidad Transición : Portabilidad, Reusabilidad, Interoperabilidad	Las 3 características tienen a su vez características de bajo nivel (7): Utilidad: Fiabilidad, Eficiencia, Ergonomía Capacidad de Mantenimiento: Facilidad de Evaluación, Comprensibilidad, Facilidad de Modificación Movilidad: Portabilidad
Se organiza usando un diagrama de árbol	Se organiza usando un diagrama de árbol

Para nuestro proyecto nos pareció más útil el modelo de McCall, debido a que su medición de calidad se enfoca en que el producto sea útil y funcional y que se mantenga así a pesar de los cambios y condiciones, en un desarrollo pequeño se debe priorizar que el software creado funcione correctamente, por lo que este modelo nos permite evaluar eso mejor que Boehm, el cual se enfoca más en la calidad del proceso más que en la del producto.

## Auditoría de la Gestión de Software

Elemento	Diagrama de Casos de Uso	Historias de Usuario
¿Se llevó a cabo la configuración?	Se usó Git, no hay otro registro externo	Se usó Git, no hay otro registro externo
¿Se realizó un versionamiento?	No	No
¿El objeto se ha modificado a lo largo del proyecto? ¿Cuántas veces?	Si, 2 veces	Si, 2 veces
¿Se han realizado revisiones técnicas para evaluar los cambios?	No	No
¿Se registraron los cambios algún documento dedicado?	No	No

Para ambos artefactos se notó que la gestión de la configuración se realizó de forma deficiente, limitándose al uso de Git sin tener un registro externo, los artefactos tuvieron ambos a lo largo del desarrollo 2 cambios cada uno, en ellos no se realizó el versionamiento, así como tampoco se llevaron a cabo revisiones por parte del Panel para evaluar los cambios y aceptarlos.

Esto se considera un riesgo y posible fuente de desorganización para el proyecto, ya que ambos artefactos son vitales para el desarrollo entero y si los cambios no se registran y notifican puede ocasionar mala comunicación entre las distintas partes del equipo que desarrollan elementos separados de los mismos artefactos, ocasionando que dichas partes no coincidan porque se desarrollan con distintas versiones de los artefactos.

Se recomienda fuertemente que se empiece con la configuración en estos artefactos para evitar riesgos futuros y verificar si no hay problemas actuales ocasionados por la falta de gestión.