

Presupuesto aproximado del proyecto – Sistema de alquiler de cabañas

Descripción general del proyecto:

El presente presupuesto corresponde al desarrollo de un sistema web para el alquiler de cabañas en la provincia de Mendoza. El desarrollo incluirá la implementación de un backend en Java con Spring Boot, una base de datos relacional para la persistencia de datos, y una interfaz web sencilla basada en Thymeleaf para la interacción de los usuarios inquilinos y dueños de cabañas.

Alcance funcional:

- ABM de cabañas para los usuarios dueños de cabañas puedan ponerlas en alquiler
- Lista las cabañas en alquiler, solo si están disponibles, y dentro de los rangos de precio, zonas y fechas seleccionados por el usuario.
- Alquiler de cabaña por parte de los usuarios inquilinos, pudiendo pagar a través de billetera virtual o tarjeta.

Estimación de esfuerzo:

El desarrollo se estima en un total aproximado de **12 horas de trabajo efectivo**, considerando tareas de codificación, configuración del proyecto, pruebas básicas y despliegue local.

El cálculo se basa en experiencias previas en proyectos similares, donde cada módulo tipo ABM o vista de consulta requiere entre una y tres horas de desarrollo efectivo.

Desglose aproximado por tipo de tarea:

ABM Cabañas: 3 horas

Consulta de cabañas en alquiler: 2 horas.

Alquiler de cabañas, con medios de pago especificados: 4 horas

Preparación y configuración inicial del proyecto: 1 hora.

Pruebas básicas y corrección de errores: 1 hora.

Deployment local y documentación mínima: 1 hora.

Total estimado: 12 horas.

Valor de referencia por hora:

De acuerdo con los valores orientativos publicados por el Colegio Profesional de Ciencias Informáticas de Córdoba (CPCIPC), disponibles en el sitio

<https://cpcipc.org.ar/honorarios-recomendados/>, la hora profesional para proyectos de desarrollo de software puede considerarse en un rango de **\$8.000 a \$15.000 ARS/hora**, dependiendo del nivel de experiencia y complejidad del proyecto.

Para esta estimación se considera un valor promedio de **\$10.000 ARS/hora**.

Presupuesto total estimado:

Escenario base (12 horas a \$10.000 ARS/hora): **\$120.000 ARS.**

Escenario profesional avanzado (12 horas a \$15.000 ARS/hora): **\$180.000 ARS.**

El costo puede variar según los ajustes requeridos por el cliente o la inclusión de funcionalidades adicionales.

Observaciones adicionales:

El proyecto se desarrollará utilizando herramientas y frameworks de uso libre (Spring Boot, Thymeleaf, y una base de datos relacional).

No se contemplan en este presupuesto costos asociados a hosting, dominio o despliegue en un entorno de producción, dado que el objetivo es la entrega de una versión funcional ejecutable en entorno local.

El presente presupuesto tiene carácter orientativo y educativo, basado en las pautas y valores de referencia sugeridos por el CPCIPC y las horas estimadas de trabajo individual.

Requisitos no funcionales – Sistema de Gestión Escolar

1. Tecnología y arquitectura:

El sistema deberá desarrollarse utilizando **Java con Spring Boot** para el backend y **Thymeleaf** para la capa de presentación, siguiendo una arquitectura **MVC (Modelo-Vista-Controlador)**.

2. Base de datos:

Los datos se almacenarán en una **base de datos relacional con MySQL o Postgres**, garantizando integridad referencial y persistencia entre sesiones.

3. Portabilidad:

El sistema deberá ser **ejecutable en entorno local**, sin depender de servicios externos ni configuraciones avanzadas de servidor.

4. Usabilidad:

La interfaz deberá ser **simple, clara y coherente**, permitiendo a usuarios no técnicos (dueños e inquilinos) realizar sus tareas sin capacitación adicional.

5. Mantenibilidad:

El código fuente deberá estar **organizado, comentado y estructurado por capas**, para facilitar futuras modificaciones o ampliaciones del sistema.

6. Rendimiento:

El sistema deberá responder en un tiempo corto para las operaciones de consulta y carga de datos bajo condiciones normales de uso.

7. Seguridad y control de acceso:

Cada usuario deberá tener acceso únicamente a las funcionalidades y datos correspondientes a su rol (por ejemplo, un usuario solo podrá modificar las cabañas que él mismo haya cargado).

8. Fiabilidad:

El sistema deberá manejar adecuadamente los errores y excepciones, evitando cierres inesperados y mostrando mensajes claros ante fallos de validación o conexión.

9. Escalabilidad básica:

El diseño deberá permitir, en una etapa posterior, incorporar nuevos módulos (como inscripciones, reportes avanzados o comunicación con tutores) sin requerir una reescritura total del código.

10. Compatibilidad:

La aplicación deberá ser accesible mediante los navegadores web más comunes (**Google Chrome, Mozilla Firefox, Microsoft Edge**) sin necesidad de instalar complementos adicionales.

11. Pruebas y calidad:

Antes de su entrega, el sistema deberá pasar por un **testing funcional básico**, verificando los flujos principales de carga y consulta de notas.

12. Documentación mínima:

El proyecto deberá incluir un breve **informe técnico** con las instrucciones de instalación, estructura del código y credenciales de ejemplo para probar los distintos roles.