

Marco de Análisis de Prompts para IA de Desarrollo Web

Criterios de Evaluación y Sistema de Puntuación (Escala de 1 a 5)

Categoría	Descripción	Criterios de Puntuación	Peso
Claridad	Nivel de claridad y ausencia de ambigüedad	1: Altamente ambiguo y confuso 2: Varios elementos poco claros 3: Mayormente claro con alguna ambigüedad 4: Claro con mínima ambigüedad 5: Perfectamente claro y sin ambigüedades	15%
Especificidad	Nivel de detalle y precisión en los requisitos	1: Extremadamente vago 2: Sólo información básica 3: Detalle moderado 4: Requisitos detallados 5: Especificaciones precisas y completas	20%
Precisión Técnica	Uso correcto de términos y conceptos técnicos	1: Errores técnicos graves 2: Varias imprecisiones técnicas 3: Problemas menores técnicos 4: Mayormente preciso 5: Totalmente preciso	15%
Provisión de Contexto	Información de fondo y contexto del proyecto	1: Sin contexto 2: Contexto mínimo 3: Contexto básico relevante 4: Buena información contextual 5: Contexto completo	10%
Calidad de la Estructura	Organización y flujo lógico del prompt	1: Estructura caótica 2: Organización deficiente 3: Estructura adecuada 4: Bien organizado 5: Estructura lógica excelente	10%
Calidad del Código Generado	Calidad del código resultante	1: Código no funcional 2: Funcional pero de baja calidad 3: Calidad aceptable 4: Buena calidad con problemas menores 5: Código excelente y listo para producción	30%

Plantilla de Documentación de Prompts

Campo	Contenido
ID del Prompt	P005-ModelToController
Fecha de Uso	03/03/2025
Modelo de IA	Claude Sonnet 3.7
Categoría de la Tarea	Backend
Tecnología	PHP
Texto Completo del Prompt	V.1 Crea un controlador para este modelo: <?php // app/Models/Stand.php namespace App\Models; use Illuminate\Database\Eloquent\Factories\HasFactory; use Illuminate\Database\Eloquent\Model; class Stand extends Model { use HasFactory; protected \$fillable = ['user_id', 'name',] }

	<pre> 'description', 'location', 'category',]; // Relaciones public function user() { return \$this->belongsTo(User::class); } public function products() { return \$this->hasMany(Product::class); } } </pre> <p>V1.1 Crea un controlador en Laravel para el siguiente modelo:</p> <pre> // app/Models/Stand.php namespace App\Models; use Illuminate\Database\Eloquent\Factories\HasFactory; use Illuminate\Database\Eloquent\Model; class Stand extends Model { use HasFactory; protected \$fillable = ['user_id', 'name', 'description', 'location', 'category',]; // Relaciones public function user() { return \$this->belongsTo(User::class); } public function products() { return \$this->hasMany(Product::class); } } </pre> <p>Requisitos del Controlador</p> <ol style="list-style-type: none"> 1. CRUD Completo: Implementa los métodos index, create, store, show, edit, update, y destroy siguiendo el patrón RESTful de Laravel. 2. Autenticación y Permisos: <ul style="list-style-type: none"> o Solo el dueño del Stand debe poder editarlo o eliminarlo. o Un usuario solo puede tener un Stand. Si ya tiene uno, debe ser redirigido a su edición. 3. Validación de Datos: <ul style="list-style-type: none"> o name: Requerido, texto, máximo 255 caracteres. o description: Opcional, texto. o location: Requerido, texto, máximo 255 caracteres. o category: Requerido, texto, máximo 255 caracteres, debe ser una de las siguientes opciones: ['Alimentación', 'Artesanía', 'Ropa', 'Productos ecológicos', 'Flores y plantas', 'Otros']. 4. Carga de Relaciones: <ul style="list-style-type: none"> o index debe listar los Stand con la relación user cargada. o show debe incluir los products disponibles (is_available = true) y la relación user. 5. Restricciones de Eliminación: Un Stand solo puede eliminarse si no tiene productos asociados. 6. Métodos Adicionales: <ul style="list-style-type: none"> o byCategory(\$category): Filtrar puestos por categoría. o search(Request \$request): Buscar puestos por name o description. <p>Entrega el controlador en PHP dentro del namespace App\Http\Controllers, siguiendo las convenciones de Laravel.</p>
Long. del Prompt	V.1 558 caracteres
	V1.1 1775 caracteres
Documentos Incluidos	No
Ejemplos Incluidos	No
Calidad de la Respuesta	V.1 5
	V.1.1 5
P. Ponderada	V.1 4
	V.1.1 5
Prob. de Código	Ninguno
Iteraciones	0
Mejoras Recomendadas	Puede mejorar proporcionando más contexto y requisitos específicos.

Tabla de Comparación de Versiones de Prompt

Versión	Camb. Clave	Cla.	Esp.	Prec. Téc.	Ctx.	Estruc.	Cal. Cód.	Punt. Pond.	Mej.
1.0	Inicial	5	3	5	2	4	5	4	-
1.1	Proporciona toda la información necesaria	5	5	5	5	5	5	5	1

Leyenda de Abreviaturas

- **Camb. Clave** = Cambios Clave
- **Cla.** = Claridad
- **Esp.** = Especificidad
- **Prec. Téc.** = Precisión Técnica
- **Ctx.** = Contexto
- **Estruc.** = Estructura
- **Cal. Cód.** = Calidad del Código
- **Punt. Pond.** = Puntuación Ponderada
- **Mej.** = Mejora