

Prueba Técnica – Desarrollador Full Stack (v2.0)

Objetivo:

Evaluar capacidad para desarrollar, integrar, desplegar y documentar soluciones full stack con los stacks requeridos, bajo condiciones similares a un entorno productivo.

Tiempo estimado: 72 horas

Formato de entrega:

- Repositorio GitHub (estructura organizada, commits descriptivos)
- Deploys funcionales accesibles
- Documentación técnica completa
- Evidencias de funcionamiento

Ejercicio 1: Sistema de Gestión de Tareas (CRUD Full Stack)

Objetivo: Evaluar dominio de Angular, Node.js, MySQL , MongoDB y despliegue integral.

Requisitos:**Frontend (Angular):**

- Implementar UI/UX funcional con Angular Material o similar
- Formularios reactivos con validaciones (ej: fecha límite > hoy)
- Filtros (por estado, responsable)
- Manejo de errores (visual feedback)

Backend (Node.js + Express):

- API REST con estructura modular (routes, controllers, services)
- ORM (Sequelize) con migraciones y seeders básicos
- Validaciones de backend (ej: campos requeridos)

- Manejo adecuado de HTTP status codes

Base de Datos:

- Modelo relacional en MySQL (diagrama ER obligatorio)
- Relaciones (ej: Tasks → Users [responsable])
- Lista de colecciones usadas MONGODB

Despliegue:

- Frontend en Netlify/Vercel o cualquier otro gratuito que permita desplegar la solución
- Backend + DB en Render/Firebase o cualquier otro gratuito que permita desplegar la solución
- Configuración de CORS y variables de entorno

Entregables:

1. Repositorio con código bien estructurado
2. URL de deploy funcional
3. Documentación:
 - Diagrama de arquitectura
 - Instrucciones de despliegue local
 - Colección Postman/Swagger (incluir casos de error)
4. Video (max 2 min) demostrando funcionalidades

Ejercicio 2: Microservicio de Autenticación JWT

Objetivo: Validar conocimientos en seguridad, MongoDB y arquitectura de servicios.

Requisitos avanzados:

- Implementar rate limiting (ej: 5 intentos de login)
- Roles básicos (admin/user) con middleware de autorización

- Refresh tokens (opcional pero valorado)
- Variables de entorno para secrets

Entregables:

1. Repositorio independiente (monorepo aceptable)
2. URL del microservicio deployado
3. Ejemplos de requests (curl/Postman)
4. Documentación:
 - Flujo de autenticación (diagrama secuencial)
 - Estructura de colecciones MongoDB
 - Consideraciones de seguridad implementadas

Ejercicio 3: Automatización de Despliegue en Linux

Objetivo: Evaluar habilidades DevOps básicas.

Requisitos ampliados:

- Script Bash que:
 - Instale Node.js, Nginx, PM2
 - Clone repositorio específico
 - Configure Nginx como reverse proxy (SSL opcional)
 - Implemente monitoreo básico con PM2
- Dockerfile funcional (opcional +5pts)

Entregables:

1. Script completo con comentarios explicativos
2. Capturas de pantalla del servidor funcionando
3. Documentación:
 - Requisitos previos del servidor

- Posibles errores y soluciones

Ejercicio 4: Ejercicio

Opción A: Integración con Notion/Trello API

- Sincronización bidireccional de tareas
- Manejo de webhooks (opcional)

Opción B: Docker Compose

- Arquitectura multicontenedor (frontend, backend, DBs)
- Script de inicialización con datos de prueba

Criterios de Evaluación

Área	Puntos
Funcionalidad	30%
Calidad de código	25%
Despliegue	20%
Documentación	15%
Seguridad	10%

Se valorará especialmente:

- Commits atómicos y mensajes descriptivos
- Variables de entorno y seguridad

- Performance básica (ej: paginación en APIs)
- Testing (aunque sea mínimo)

enviar entrega de prueba para su revisión a el correo

sistemas@prevrenal.org