

Reto Técnico — Optimización, i18n (A11y) y Pruebas con Jest en Angular

Fecha límite de entrega: **25 de septiembre de 2025 (America/Bogota)**

Repositorio base: **<https://github.com/caalzate91/angular-optimization-accebility>**

Rama inicial de trabajo: **feature/unit-test**

1. Introducción

Este reto consolida lo visto en los laboratorios: optimización con NgOptimizedImage, usabilidad/accesibilidad con i18n mediante Angular Localize y pruebas automatizadas con Jest. El objetivo es producir evidencia objetiva (Lighthouse, capturas y pruebas) listo para revisión.

2. Instrucciones generales

1) Realiza un fork del repositorio base y trabaja en tu fork (individual).

URL: <https://github.com/caalzate91/angular-optimization-accebility>

Rama: feature/unit-test

2) En tu fork, usa la rama base indicada y crea tu rama de trabajo:

```
git checkout feature/unit-test
```

3) Enviar un PR al repositorio base a través de sus repositorios trabajados en el fork.

3. Requisitos del reto

A) Optimización con NgOptimizedImage

- Identifica la imagen que más impacta el LCP de la vista inicial.
- Reemplaza su `` por la directiva `NgOptimizedImage`, marca la imagen LCP con `priority` y define `width/height` reales.

- Para imágenes no críticas, aplica lazy-load.
- Ejecuta Lighthouse antes/después y adjunta evidencias (capturas).

B) Usabilidad y Accesibilidad con i18n (Angular Localize)

- Ajusta la aplicación para que detecte el idioma y que se pueda cambiar a otro idioma.

C) Pruebas con Jest — 3 pruebas nuevas

Escribe 3 pruebas adicionales (pueden ser unitarias o de integración ligera) las cuales se sugieren los siguientes :

- 1) Servicio con HTTP: caso de éxito (y opcional error) verificando método/URL y respuesta simulada.
- 2) Componente: interacción del usuario y verificación del DOM con selectores accesibles.
- 3) Integración ligera: componente + servicio + HTTP mock.

4. Entregables

- Evidencias: capturas de Lighthouse (antes/después), resultados de pruebas y notas técnicas en README.

5. Criterios de evaluación (Rúbrica)

| Área | Porcentaje | Qué se evalúa |
|-------------------------|------------|--|
| Optimización | 30 | Uso correcto de NgOptimizedImage en LCP (priority, width/height), lazy en no críticas y evidencia en Lighthouse. |
| i18n (A11y/Usabilidad) | 30 | Cambios notorios de textos en títulos y descripciones en la página web |
| Pruebas con Jest | 30 | Tres pruebas nuevas (unitarias y/o de integración). |
| Documentación/Evidencia | 10 | README y capturas claras; explicación de decisiones técnicas. |

6. Sugerencias y ayudas

- Jest usa JSDOM; algunos comportamientos visuales pueden diferir del navegador real.
- Prioriza consultas accesibles (role, name, aria-*) en pruebas de componentes.
- Evita flakiness: usa async/await o fakeAsync según convenga.
- Verifica method/URL/body en HTTP y usa verify() del HttpTestingController.

7. Referencias (lectura recomendada)

- **NgOptimizedImage** — **guía**: <https://angular.dev/guide/image-optimization>
- **NgOptimizedImage** — **API**: <https://angular.dev/api/common/NgOptimizedImage>
- **i18n** — **guía general**: <https://angular.dev/guide/i18n>
- **registerLocaleData** — **API**: <https://angular.dev/api/common/registerLocaleData>
- **HTTP testing** — **guía (provideHttpClientTesting)**: <https://angular.dev/guide/http/testing>
- **jest-preset-angular** — **instalación**:
<https://thymikee.github.io/jest-preset-angular/docs/getting-started/installation>
- **jest-preset-angular** — **repositorio**: <https://github.com/thymikee/jest-preset-angular>
- **Jest** — **configuración**: <https://jestjs.io/docs/configuration>

Fecha máxima de entrega: **25 de septiembre de 2025**