



Parte del Plan de Trabajo de Servicio Comunitario

## **Sistema ANAR**

Propuesta: Estrategia de Pruebas

Propuesta elaborada por:  
Arturo Yepez | 15-11551  
Estudiante Ing. de la Computación



# Introducción

## Propósito del Documento

Este documento servirá como documentación y propuesta para la finalización de las pruebas que han de desarrollarse para el Sistema ANAR, creado dentro de un marco de proyecto de Servicio Comunitario.

Inicialmente, las pruebas estaban diseñadas para ser cubiertas dentro del alcance original de la propuesta, pero dado los cambios de alcance y las dificultades encontradas durante el desarrollo original, estas pruebas quedaron por fuera de la implementación.

Este documento de especificación sirve de base para los desarrolladores, proporcionándoles los detalles necesarios para el desarrollo futuro de las pruebas que cubran todos los módulos desarrollados, bajo la idea original.

## Objetivos del Producto Mínimo Viable (MVP)

Los objetivos y beneficios obtenidos luego de finalizado el desarrollo son al menos los siguientes:

- La introducción de una librería de pruebas unitarias.
- Desarrollo de pruebas unitarias en los módulos donde tenga sentido
- Investigación sobre el desarrollo de pruebas de integración
- Investigación sobre el desarrollo de pruebas end-to-end, cuando se haya hecho la integración del Sistema ANAR a uno de los proyectos web
  - Estas deberán de ser desarrolladas para cada una de las aplicaciones que existan dentro del Sistema ANAR



# Pruebas

## Librería

Para la elaboración de las pruebas, se propone la siguiente librería:

Nombre	Descripción
<b>Jest</b>	Jest es una librería que ofrece un marco de pruebas de JavaScript con un enfoque en la simplicidad.

Más información de la librería se puede encontrar en el siguiente [enlace](#).

Jest es una excelente opción para pruebas unitarias debido a varias razones clave. En primer lugar, su sintaxis simple y su configuración mínima hacen que sea fácil de aprender y usar, lo que ahorra tiempo y esfuerzo en la configuración inicial. Además, Jest proporciona una amplia gama de funcionalidades integradas, como mocks y spies, que facilitan la simulación de dependencias y la verificación del comportamiento del código. Otra ventaja importante es su velocidad de ejecución, ya que utiliza técnicas como la ejecución en paralelo para optimizar el tiempo de prueba, lo que es fundamental en proyectos con crecimiento.

Una de las características destacadas que ofrece Jest como librería de pruebas unitarias es su capacidad para agregar cobertura de código de manera sencilla y efectiva. Esta funcionalidad integrada te permite evaluar qué porcentaje del código fuente está siendo cubierto por tus pruebas, lo cual es fundamental para asegurar la calidad y la efectividad de tu suite de pruebas. Además, Jest ofrece opciones de configuración flexibles para ajustar la cobertura según tus necesidades específicas, como la inclusión o exclusión de directorios o archivos particulares en el informe de cobertura. Esto hace que Jest sea una herramienta completa para no solo realizar pruebas unitarias, sino también para evaluar y mejorar la cobertura de código en el proyecto del Sistema ANAR..

## Enfoque de Implementación

Se recomienda hacer un enfoque de implementación que siga los siguientes pasos:

1. Instalación de la librería
2. Diseño de estándares para la implementación



- a. Ubicación de pruebas
  - b. Configuración básica
  - c. Decidir si incluir Cobertura de código
3. Desarrollo de pruebas

Para el desarrollo de pruebas, se debe de tener en cuenta que al inicio no todos los controladores y módulos van a poder hacerse pruebas dado que muchas de las pruebas que los impactan serían consideradas como pruebas de integración dado las distintas conexiones que realizan (como base de datos, por ejemplo).

De esta forma, se exhorta a que las pruebas de integración sean consideradas una vez la implementación de pruebas unitarias sea realizada.