



TECNOLÓGICO
NACIONAL DE MEXICO



**Instituto Tecnológico
de Ciudad Victoria**

INGENIERIA EN SOFTWARE

REPORTE DE CALIDAD DE SOFTWARE

*Ing . Osvaldo Daniel Fernandez
Bonilla*

Realizado por Dariela Eleida Gonzalez Villanueva

Entregado el 04 de Diciembre del 2025

Indice

Descripción del Software y Estándares.....	3
Módulos Funcionales Clave:.....	3
Factores de Calidad y Métricas Aplicadas.....	4
a) Eficacia / Adecuación Funcional.....	4
b) Rendimiento / Eficiencia.....	4
c) Fiabilidad.....	4
d) Usabilidad.....	4
Analisis Técnico y Rendimiento (Métricas de Código y Datos).....	5
1. Calidad y Mantenibilidad del Código (C# / ASP.NET Core).....	5
2. Rendimiento y Eficiencia del Sistema.....	5

Descripción del Software y Estándares

El proyecto **Lit-Match: Cita a Ciegas con Libros** se establece como una **plataforma de suscripción mensual** y un proyecto de Ingeniería de Software cuyo valor reside en la tecnología de recomendación que implementa. Su objetivo principal es abordar la **fatiga de la elección** y la ineficacia de los sistemas de recomendación existentes, los cuales perpetúan una "burbuja de confort" al basarse en métricas superficiales como el género y la popularidad.

La solución reintroduce la sorpresa y la emoción mediante un enfoque de "Cita a Ciegas" con un libro físico.

La solución reintroduce la sorpresa y la emoción mediante un enfoque de "Cita a Ciegas" con un libro físico.

Módulos Funcionales Clave:

- **Sistema Central de Suscripción y Gestión:** Módulo para la **Adquisición y Registro de Usuarios** y la **Gestión de Suscripciones** mensuales, administrando la estructura de ingresos.
- **Algoritmo de Matching Avanzado (AMA):** El activo de propiedad intelectual del proyecto. Utiliza el **Procesamiento de Lenguaje Natural (NLP)** y **Machine Learning** para:
 - Administrar el **Cuestionario de Estilo (Input)** del lector.
 - Asignar "**huellas**" emocionales, estilísticas y temáticas a las obras literarias (Corpus) y a las respuestas del lector.
 - Realizar el **Matching Emocional** para conectar lectores con libros basándose en la experiencia de lectura deseada.
- **Logística y Marketplace:** Responsable de generar el "**misterioso Envío (Output)**". También funciona como una plataforma de lanzamiento objetiva y equitativa para autores y talento emergente, permitiendo al Autor/Editor **Registrar Obra** y **Recibir Datos de Visibilidad**.
- **Sistema de Feedback:** Permite al usuario dar su opinión después de recibir el libro, lo que ayuda a **refinar futuras recomendaciones** (retroalimentación continua para el AMA).

Factores de Calidad y Métricas Aplicadas

El valor fundamental del proyecto **Lit-Match** reside en su tecnología central: el Algoritmo de Matching Avanzado (AMA). Por lo tanto, los factores de calidad se centran en verificar y medir la eficacia, el rendimiento y la fiabilidad del sistema para asegurar la viabilidad tecnológica y el éxito financiero.

a) Eficacia / Adecuación Funcional

Este es el factor más crítico. Mide si el sistema cumple con su propósito de negocio principal: realizar un *matching* de alta calidad que supere a las recomendaciones superficiales.

- **Requisito No Funcional Objetivo: Precisión del Algoritmo.** El sistema debe garantizar una alta tasa de acierto en la afinidad lectora para reintroducir la emoción del descubrimiento y diferenciar el servicio.

b) Rendimiento / Eficiencia

Mide la capacidad del sistema para cumplir los objetivos de escalabilidad y velocidad bajo las restricciones de implementación.

- **Requisito No Funcional Objetivo: Escalabilidad y Tiempo de Respuesta.** El sistema debe ser capaz de procesar eficientemente el Corpus de obras y el Cuestionario de Estilo de miles de suscriptores sin degradación del servicio.

c) Fiabilidad

Mide la capacidad del sistema para mantener un nivel específico de rendimiento cuando se utiliza en condiciones establecidas.

- **Requisito No Funcional Objetivo: Disponibilidad del Servicio de Suscripción.** La plataforma opera con ingresos recurrentes; cualquier fallo en los módulos de pago o registro de usuarios afecta directamente la visibilidad financiera y el punto de equilibrio.

d) Usabilidad

Mide la facilidad con la que el usuario puede interactuar con el sistema para lograr sus objetivos.

- **Requisito No Funcional Objetivo: Facilidad de Uso del Input/Feedback.** El lector debe poder completar el Cuestionario de Estilo y el Sistema de Feedback de forma sencilla para garantizar que el AMA reciba datos de alta calidad.

Analisis Técnico y Rendimiento (Métricas de Código y Datos)

Dado que el proyecto Lit-Match se encuentra en una fase de planificación y el valor principal reside en la tecnología de *backend* (el AMA y la gestión de datos), este análisis se centra en las métricas esenciales para asegurar la escalabilidad, la mantenibilidad y la eficiencia de costos del código C# y la base de datos SQL Server.

1. Calidad y Mantenibilidad del Código (C# / ASP.NET Core)

La calidad del código es fundamental para la viabilidad a largo plazo, especialmente en el **Algoritmo de Matching Avanzado (AMA)**, que será la parte más compleja y que más se modificará.

- **Complejidad Ciclomática (CC):** Mediremos la complejidad de las funciones en el código C#. El objetivo es mantener una **Complejidad Ciclomática baja (idealmente, menor a 10)** en todos los métodos críticos del AMA y de los controladores de suscripción. Esto asegura que la lógica sea fácil de leer, probar y depurar por cualquier miembro del equipo.
- **Cobertura de Pruebas Unitarias:** Se exige que un alto porcentaje del código de la **Capa de Lógica de Negocio** y el AMA esté cubierto por pruebas automatizadas. El umbral mínimo es del **85%**. Esto garantiza que las funciones esenciales (como el cálculo de *matching* o el registro de pagos) seguirán funcionando correctamente incluso después de actualizaciones o modificaciones al código.
- **Desacoplamiento (Coupling):** Utilizaremos **Interfaces** y la Inyección de Dependencias de ASP.NET Core para asegurar un **acoplamiento bajo** entre la lógica de negocio y la capa de acceso a datos (Entity Framework Core). Esto permite, por ejemplo, cambiar el proveedor de la base de datos o el servicio de Machine Learning sin tener que reescribir toda la aplicación.

2. Rendimiento y Eficiencia del Sistema

El rendimiento se mide en la rapidez con la que el sistema puede operar y procesar grandes volúmenes de datos, lo que influye directamente en los costos operativos y la experiencia del usuario.

- **Latencia de API (AMA Execution Time):** Es el tiempo crítico que tarda el sistema en comunicarse con los servicios de **NLP/Machine Learning** para generar una Huella Emocional del lector o procesar un libro. El objetivo es mantener este tiempo **por debajo de 1 segundo**. Esto es vital porque afecta directamente el **Costo de ML por Suscriptor** (al ser un servicio de pago por uso) y la velocidad de respuesta al usuario.
- **Tiempo de Respuesta de Consulta (DB Query Time):** Se evaluará el tiempo que tarda la base de datos **SQL Server** en ejecutar las consultas más complejas, como la selección de obras elegibles por criterios de huella. El objetivo es que las consultas críticas tarden **menos de 200 milisegundos**. Un buen rendimiento de la base de datos es esencial para la escalabilidad del sistema.

- **Utilización de CPU:** Se monitoreará el uso de recursos del servidor de aplicaciones. Se buscará que el uso promedio de CPU se mantenga entre el **50% y 70%** bajo una carga de trabajo alta. Esto indica que el sistema está bien optimizado y tiene capacidad de reserva para absorber picos de tráfico.
- **Tasa de Errores por Segundo:** Medida de **Fiabilidad**. Se debe mantener una tasa de fallos de conexión con servicios externos (pagos, ML) o errores internos del servidor **por debajo del 0.1%**.

