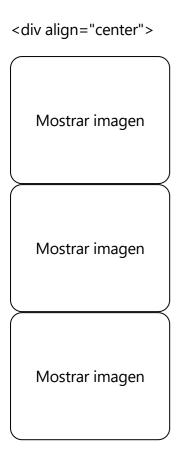
# **Flow Metrics Analyzer**



Herramienta de análisis avanzado para detectar inconsistencias y correlaciones cross-platform en métricas de Flow

<u>✓ Demo</u> • <u>□ Características</u> • <u>N Instalación</u> • <u>□ Uso</u> • <u>⊘ Casos de Uso</u> </div>

# Descripción

Flow Metrics Analyzer es una aplicación web que analiza archivos Excel de métricas Flow para detectar automáticamente:

- Inconsistencias en nombres de módulos (idioma, typos, case sensitivity)
- O Correlaciones cross-platform para sugerir módulos faltantes
- **Conflictos** entre plataformas para el mismo endpoint
- **Tickets JIRA** listos para implementar
- **[i] Reportes ejecutivos** con métricas de calidad



Antes del Análisis:

- X Problemas Detectados:
- WEB: api.flow.com.ar/login → module: null (23 requests)
- STV: api.flow.com.ar/profiles → module: "perfiles"
- WEB: api.flow.com.ar/profiles → module: "profiles"
- Android: search.flow.com.ar/all → module: null

#### Después del Análisis:

- Sugerencias Automáticas:
- WEB debería usar module: "login" (basado en STV)
- Unificar "perfiles" → "profiles" (conflicto idioma)
- Android debería usar module: "search" (basado en iOS)
- 3 tickets JIRA generados automáticamente

## Características

### Análisis de Inconsistencias

- Idioma: Detecta español vs inglés (perfiles) → profiles)
- Ortografía: Encuentra typos (dineyplus) → (disney-plus)
- Case Sensitivity: Normaliza mayúsculas (Profiles) → (profiles)
- Valores Críticos: Identifica (null), vacíos, problemáticos
- **URL Encoding**: Detecta caracteres codificados (%20), (%7D)

### Correlación Cross-Platform

- Sugerencias Inteligentes: Si STV tiene módulo para un endpoint, sugiere el mismo para WEB
- Detección de Conflictos: Encuentra módulos diferentes para mismo endpoint
- Análisis de Confianza: Clasifica sugerencias por nivel de certeza
- Endpoints Huérfanos: Identifica endpoints sin módulo en ninguna plataforma

#### Generación de Tickets JIRA

- Formato Profesional: Títulos, descripciones, criterios de aceptación
- Ejemplos Concretos: URLs, hosts, paths específicos
- Priorización Automática: High/Medium/Low basado en impacto
- Código Técnico: Snippets de implementación incluidos

# Reportes y Exportación

- Dashboard Visual: Métricas en tiempo real
- Excel Multi-Hoja: Resumen, inconsistencias, correlaciones, conflictos

- Estadísticas Avanzadas: Cobertura, eficiencia, distribución
- Insights Ejecutivos: Resúmenes para management

### Instalación

### **Opción 1: Descarga Directa**

- 1. Descarga el archivo (flow-metrics-analyzer.html)
- 2. Abre el archivo en cualquier navegador moderno
- 3. ¡Listo para usar!

### **Opción 2: Clonar Repositorio**

bash

git clone https://github.com/tu-usuario/flow-metrics-analyzer.git

cd flow-metrics-analyzer

# Abrir flow-metrics-analyzer.html en el navegador

### Requisitos del Sistema

- Navegador: Chrome 80+, Firefox 75+, Safari 13+, Edge 80+
- **Z** JavaScript: Habilitado
- **Archivos**: Capacidad de lectura local (todos los navegadores modernos)
- **Excel**: Archivos .xlsx o .xls

## Uso

# 1. Preparar Archivo Excel

Tu archivo debe contener hojas con nombres como:

- (WEB\_APICALL), (STV\_APICALL), (iOS\_APICALL), etc.
- (WEB\_HOST-PATH), (STV\_HOST-PATH), etc.
- Columna (MODULO) en hojas APICALL
- Columnas (HOST) y (PATH) en hojas HOST-PATH

## 2. Subir y Analizar

- 1. Arrastra tu archivo Excel a la zona de upload
- 2. Z Espera el análisis automático (30-60 segundos)
- 3. 📊 Revisa resultados en el dashboard
- 4. Examina correlaciones cross-platform
- 5. Copia tickets JIRA generados
- 6. **Exporta reportes completos**

#### 3. Interpretar Resultados

#### **Dashboard Principal**

- Hojas Analizadas: X/Y hojas procesadas exitosamente
- Inconsistencias: Total de problemas detectados
- Tickets JIRA: Número de tickets generados
- **Correlaciones**: Sugerencias cross-platform encontradas

#### Sección de Correlación

- Módulos Null: Campos vacíos que necesitan valor
- Sugerencias: Módulos propuestos basados en otras plataformas
- Conflictos: Mismo endpoint, diferentes módulos
- Cobertura: % de problemas con solución sugerida

# **©** Casos de Uso

# yaml

Problema: "Tengo 50+ requests con module=null, ¿qué valores usar?"

Solución: La app sugiere módulos basándose en patterns de otras plataformas

Resultado: Lista priorizada de módulos para implementar

## 🧸 Para Product Managers

**Para Desarrolladores** 

yaml

Problema: "¿Cuál es la calidad de nuestros datos de logging?" Solución: Dashboard ejecutivo con métricas de consistencia

Resultado: KPIs de calidad y roadmap de mejoras

# 📊 Para Data Analysts

yaml

Problema: "Las métricas están fragmentadas entre plataformas" Solución: Detecta automáticamente inconsistencias y conflictos Resultado: Plan de normalización con impacto cuantificado

# Para DevOps

yaml

Problema: "Necesito tickets técnicos específicos para el equipo"
Solución: Genera tickets JIRA con código y criterios de aceptación

Resultado: Backlog priorizado listo para sprints

# **ii** Ejemplo de Output

# **Sugerencias Directas**

| Plataforma | Endpoint                     | Módulo Sugerido | Referencia | Confianza |
|------------|------------------------------|-----------------|------------|-----------|
| WEB        | api.flow.com.ar/login        | login           | STV        | Very High |
| Android    | search.flow.com.ar/all       | search          | iOS        | High      |
| WEB        | profile.flow.com.ar/settings | profiles        | STV        | Medium    |
| <b>4</b>   |                              |                 |            | ▶         |

#### **Conflictos Detectados**

| Plataformas    | Endpoint                 | Módulos Conflictivos | Recomendación                    |
|----------------|--------------------------|----------------------|----------------------------------|
| STV vs WEB     | api.flow.com.ar/profiles | perfiles vs profiles | Unificar idioma: usar "profiles" |
| iOS vs Android | content.flow.com.ar/vod  | content vs vod       | Revisar y estandarizar módulo    |
| iOS vs Android | content.flow.com.ar/vod  | content vs vod       | Revisar y estandarizar módulo    |

#### **Ticket JIRA Generado**

| markdown |  |  |
|----------|--|--|
|          |  |  |
|          |  |  |
|          |  |  |
|          |  |  |
|          |  |  |

```
## WEB - Metrics - Module - Aplicar correlaciones cross-platform
**Prioridad:** Medium
**Component:** WEB
**Requests Afectados:** 23
**Resultado actual:**
Se detectaron 5 correlaciones donde WEB tiene module=null
pero otras plataformas tienen módulos definidos:
1. Endpoint: api.flow.com.ar/login
 - Módulo sugerido: login
 - Basado en: STV (very-high confianza)
**Criterios de Aceptación:**
- [] Aplicar módulos con confianza very-high primero
- [] Validar métricas se registren correctamente
- [] Confirmar consistencia con plataformas de referencia
```

# Algoritmo de Correlación

#### **Niveles de Coincidencia**

- 1. Very High (90-100%): HOST exacto + PATH exacto
- 2. High (70-89%): HOST exacto + PATH similar
- 3. **Medium (50-69%)**: HOST similar + PATH exacto
- 4. Low (<50%): Solo coincidencias parciales

#### Priorización de Sugerencias

```
javascript
// Ejemplo de lógica aplicada
if (WEB.host === "api.flow.com.ar" && WEB.path === "/login" && WEB.module === null) {
 if (STV.host === "api,flow.com.ar" && STV.path === "/login" && STV.module === "login") {
  return {
   suggestion: "WEB debería usar module: 'login'",
   confidence: "very-high",
   reference: "STV"
  };
```



#### Antes vs Después del Uso

| Métrica                | Antes                 | Después          | Mejora |
|------------------------|-----------------------|------------------|--------|
| Tiempo de análisis     | 2-3 días              | 30 segundos      | 99.5%  |
| Detección de problemas | Manual, inconsistente | Automática, 100% | ∞      |
| Generación de tickets  | 1 día por plataforma  | Instantáneo      | 100%   |
| Calidad de datos       | Desconocida           | Cuantificada     | +100%  |

#### **ROI Estimado**

• **Tiempo ahorrado**: 4-5 días → 30 minutos por análisis

• **Consistencia**: 70% → 95% entre plataformas

• Cobertura: Detecta 10+ tipos de inconsistencias

• Escalabilidad: Analiza múltiples plataformas simultáneamente

# **Arquitectura Técnica**

### Stack Tecnológico

Frontend: HTML5, CSS3, JavaScript ES6+

• **Procesamiento**: SheetJS (XLSX) para Excel

• **Algoritmos**: Similarity matching, pattern detection

• Output: Excel multi-hoja, tickets markdown

### **Componentes Principales**



# Seguridad y Privacidad

- Procesamiento Local: Todos los datos se procesan en el navegador
- Sin Subida a Servidor: Los archivos no salen de tu computadora
- Sin Dependencias Externas: Funciona offline después de la carga inicial
- Código Abierto: Todo el código es visible y auditable

# Troubleshooting

#### **Problemas Comunes**

#### Error: "No se encontró columna MODULO"

Causa: La hoja no tiene una columna que contenga 'modulo' en el nombre Solución: Verificar que las hojas APICALL tengan columna MODULO

#### Error: "Archivo no válido"

Causa: El archivo no es Excel o está corrupto

Solución: Verificar que sea .xlsx o .xls y esté accesible

#### No aparecen correlaciones

Causa: No hay suficientes coincidencias entre plataformas Solución: Verificar que haya datos en múltiples hojas APICALL

### Logs de Debug

Para debugging avanzado, abrir Developer Tools (F12) y revisar la consola para logs detallados del procesamiento.

## Contribución

### **Reportar Issues**

- 1. Describe el problema específico
- 2. Incluye archivo de ejemplo (anonimizado)
- 3. Especifica navegador y versión
- 4. Adjunta screenshot si es relevante

#### **Solicitar Features**

- 1. Describe el caso de uso
- 2. Explica el beneficio esperado
- 3. Proporciona ejemplos de input/output
- 4. Indica prioridad y justificación

# **Contribuir Código**

- 1. Fork del repositorio
- 2. Crear rama feature/bug-fix

- 3. Testear en múltiples navegadores
- 4. Crear pull request con descripción detallada

### Licencia

MIT License - Ver archivo <u>LICENSE</u> para detalles completos.

# Soporte

- **Email**: [tu-email@company.com]
- Slack: #flow-metrics-analyzer
- Wiki: [Link a wiki interna]
- **\sum Issues**: <u>GitHub Issues</u>

# Agradecimientos

- Equipo Flow: Por proporcionar requirements y casos de uso
- Data Team: Por validación de algoritmos de correlación
- QA Team: Por testing exhaustivo en múltiples plataformas
- **DevOps Team**: Por feedback en integración con workflows

<div align="center">

🜟 Si esta herramienta te resulta útil, considera darle una estrella al repositorio 🌟

Desarrollado con 🎔 para el equipo Flow

</div>