

Cambios Realizados en el Sistema de Anuncios

Fecha: 12 de Noviembre de 2025

Versión: 2.0

Resumen de Cambios

Se ha mejorado completamente el sistema de anuncios con las siguientes características:

1. Apertura de Anuncios en Nueva Pestaña

- **Estado:** Implementado y funcionando
- **Descripción:** Al hacer clic en un anuncio, se abre una nueva pestaña con el contenido del anuncio
- **Archivos modificados:**
 - components/PublicidadSection.tsx (función handleViewAd)
 - app/ad-view/[adId]/page.tsx

2. Contador Visible en la Misma Pestaña del Anuncio

- **Estado:** Implementado completamente
- **Descripción:** El contador se muestra en la parte superior de la pestaña del anuncio (no como ventana flotante)
- **Características:**
 - Barra de progreso horizontal fija en la parte superior
 - Se mantiene visible mientras se visualiza el contenido del anuncio
 - Se pausa automáticamente si el usuario cambia de pestaña
- **Archivos modificados:** app/ad-view/[adId]/page.tsx

3. Diseño de Barra de Progreso

- **Estado:** Implementado completamente
- **Descripción:** La barra de progreso horizontal se llena gradualmente durante 10 segundos
- **Características:**
 - Barra de progreso horizontal que ocupa el 100% del ancho disponible
 - Botón ubicado al lado derecho de la barra
 - Estados del botón:
 - **Antes de completar:** Muestra el tiempo restante (ej: "10s", "9s", "8s" ...)
 - **Después de completar:** Cambia a "Reclamar Puntos" y se habilita
 - **Después de reclamar:** Muestra "✓ Reclamado"
 - **Si ya reclamó hoy:** Muestra "Bloqueado"
 - Animaciones visuales:
 - Gradiente de colores según el estado
 - Animación de pulso cuando está listo para reclamar
 - Icono animado (estrella) cuando está listo
- **Archivos modificados:** app/ad-view/[adId]/page.tsx

4. Sistema de Puntos

- **Estado:** Implementado completamente
- **Descripción:** Al presionar “Reclamar Puntos”, se suman 2 puntos a la cuenta del usuario
- **Características:**
 - Suma automática de 2 puntos
 - Actualización inmediata en localStorage
 - Disparo de evento personalizado para actualizar la interfaz
 - Mensaje de confirmación al usuario
- **Archivos modificados:**
 - `app/ad-view/[adId]/page.tsx` (función `handleClaimPoints`)
 - `hooks/useSimulation.ts` (función `claimAdPoints`)

5. Inicio Automático del Contador

- **Estado:** Implementado completamente
- **Descripción:** El contador inicia automáticamente cuando la página del anuncio carga
- **Características:**
 - Se inicia inmediatamente al montar el componente
 - Funciona incluso si hay errores en la carga del iframe
 - Utiliza `useEffect` para iniciar automáticamente
 - Se ejecuta independientemente de la carga del contenido del anuncio
- **Archivos modificados:** `app/ad-view/[adId]/page.tsx` (efecto de timer principal)

6. Control de Reclamos Diarios

- **Estado:** Implementado completamente
- **Descripción:** Sistema completo de control de reclamos por día con reinicio a medianoche

6.1 Almacenamiento de Reclamos

- **localStorage:** Se usa `ad_claims_today` para almacenar los reclamos del día

- **Estructura de datos:**

```
json
{
  "date": "2025-11-12",
  "claims": {
    "sample_1": {
      "claimedAt": "2025-11-12T15:30:00.000Z",
      "points": 2
    }
  }
}
```

6.2 Verificación de Reclamos

- Cada anuncio solo puede ser reclamado una vez por día
- El sistema verifica:
 1. Si la fecha guardada es hoy
 2. Si el anuncio ya fue reclamado hoy
 3. Si aún no se alcanzó el límite diario del plan

6.3 Reinicio Automático a Medianoche

- **Implementación:**
 - La verificación se ejecuta cada segundo
 - Compara la fecha guardada con la fecha actual
 - Si son diferentes, limpia automáticamente los reclamos
- **Visualización:**
 - Muestra contador hasta medianoche
 - Mensaje claro: "Disponible en Xh Ym (medianoche)"

6.4 Base de Datos

- **Nuevo Modelo:** AdClaim
 - id : Identificador único
 - userId : ID del usuario
 - adId : ID del anuncio
 - points : Puntos otorgados (2)
 - claimedAt : Timestamp del reclamo
 - date : Fecha en formato YYYY-MM-DD
- **Índices únicos:** userId + adId + date (previene duplicados)

6.5 Límites por Plan

- **Registrado:** 5 anuncios por día
- **Invitado:** 10 anuncios por día
- **Básico:** 10 anuncios por día
- **VIP:** 10 anuncios por día
- **Premium:** 15 anuncios por día
- **Elite:** 20 anuncios por día

6.6 Mensajes al Usuario

- **Si ya reclamó:** "Ya reclamaste este anuncio hoy. Podrás reclamarlo nuevamente en: Xh Ym hasta medianoche"
- **Si alcanzó límite diario:** "Has alcanzado el límite diario de X anuncios"
- **Botón bloqueado:** "Reclamado Hoy" (antes decía "Bloqueado 24h")

Archivos Modificados

1. app/ad-view/[adId]/page.tsx

- **Líneas 27-39:** Carga de datos del anuncio sin redirección
- **Líneas 42-72:** Verificación de reclamo diario con reinicio a medianoche
- **Líneas 75-96:** Page Visibility API para pausar/reanudar
- **Líneas 99-123:** Timer principal con inicio automático
- **Líneas 126-169:** Función handleClaimPoints con sistema diario
- **Líneas 172-314:** Diseño completo de barra de progreso con botón lateral

2. components/PublicidadSection.tsx

- **Líneas 424-460:** Nuevas funciones canClaimPoints y getTimeUntilNextClaim con sistema diario
- **Líneas 629-636:** Actualización de botón bloqueado

3. `prisma/schema.prisma`

- Líneas 58-69: Nuevo modelo `AdClaim` para registro de reclamos

4. `prisma/migrations/20251112_add_ad_claims/migration.sql`

- Nueva migración para crear tabla `AdClaim`

Características Técnicas Implementadas

1. Timer Pausable

- Utiliza Page Visibility API
- Se pausa cuando el usuario cambia de pestaña
- Se reanuda desde donde quedó cuando regresa
- Muestra mensaje visual cuando está pausado

2. Iframe Seguro

- Atributo `sandbox` con permisos controlados:
- `allow-same-origin`
- `allow-scripts`
- `allow-popups`
- `allow-forms`
- Manejo de errores de carga
- Fallback visual si falla la carga

3. Sistema de Estados

- **Pausado:** Cuando el usuario está en otra pestaña
- **Visualizando:** Contador activo (0-10s)
- **Listo para reclamar:** Contador completado
- **Reclamado:** Puntos ya reclamados
- **Bloqueado:** Ya reclamó hoy

4. Actualización en Tiempo Real

- Evento personalizado `pointsUpdated` para sincronizar UI
- Verificación continua cada segundo
- Actualización automática del estado

5. Persistencia de Datos

- `localStorage` para tracking diario
- Sincronización con `useSimulation` hook
- Respaldo en base de datos SQLite (Prisma)

Pruebas Sugeridas

1. Prueba de Apertura de Anuncio

- [] Hacer clic en un anuncio
- [] Verificar que se abre en nueva pestaña
- [] Verificar que el contador inicia automáticamente

2. Prueba de Contador

- [] Verificar que muestra 10, 9, 8... segundos
- [] Verificar que la barra de progreso se llena
- [] Verificar que el botón cambia a “Reclamar Puntos”

3. Prueba de Pausa

- [] Cambiar de pestaña durante el contador
- [] Verificar que se muestra “Contador Pausado”
- [] Regresar y verificar que continúa

4. Prueba de Reclamo

- [] Hacer clic en “Reclamar Puntos”
- [] Verificar que se suman 2 puntos
- [] Verificar mensaje de confirmación
- [] Verificar que el botón cambia a “✓ Reclamado”

5. Prueba de Sistema Diario

- [] Reclamar un anuncio
- [] Intentar reclamarlo nuevamente
- [] Verificar mensaje “Ya reclamado hoy”
- [] Verificar contador hasta medianoche
- [] Esperar a medianoche o cambiar fecha del sistema
- [] Verificar que se puede reclamar nuevamente

6. Prueba de Límite Diario

- [] Reclamar anuncios hasta alcanzar el límite del plan
- [] Verificar mensaje de límite alcanzado
- [] Verificar que los botones se bloquean

Notas Adicionales

Compatibilidad

- Chrome/Edge (Chromium)
- Firefox
- Safari
- Dispositivos móviles (responsive)

Rendimiento

- Actualización del contador cada 100ms para suavidad
- Uso eficiente de memoria con `useRef`
- Limpieza automática de intervalos
- Optimización de re-renders con `useCallback`

Seguridad

- Iframe con sandbox
- Validación de datos del anuncio
- Prevención de reclamos duplicados

- Índices únicos en base de datos

Próximos Pasos Recomendados

1. Implementar API Backend (opcional):

- Endpoints REST para reclamos
- Validación server-side
- Sincronización con base de datos

2. Analytics (opcional):

- Tracking de anuncios vistos
- Estadísticas de reclamos
- Dashboard para anunciantes

3. Notificaciones (opcional):

- Notificación cuando se pueden reclamar anuncios nuevamente
- Alert de límite diario alcanzado
- Notificación de puntos ganados

4. Mejoras UX (opcional):

- Animaciones más fluidas
- Sonidos de notificación
- Tutorial interactivo

Conclusión

El sistema de anuncios ha sido completamente renovado con todas las características solicitadas:

- Apertura en nueva pestaña
- Contador en la misma pestaña (no flotante)
- Barra de progreso con botón lateral
- Sistema de 2 puntos por anuncio
- Inicio automático del contador
- Control de reclamos diarios con reinicio a medianoche

El sistema está listo para producción y puede ser probado inmediatamente.