Academia Española de Neurocirugía

Sistema de Alertas Automáticas de Oposiciones/MIR de Neurocirugía

Documento de Especificación Técnica y Estrategia de Implementación

Versión 1.0

29 de Marzo de 2026

Destinatarios: Academia Española de Neurocirugía (AEN)

Tabla de Contenidos

Sistema de Alertas Automáticas de Oposiciones/MIR de Neurocirugía

Academia Española de Neurocirugía (AEN)

Documento de Especificación Técnica y Estrategia de Implementación

Versión: 1.0

Fecha: 29 de Marzo de 2026

Destinatarios: Academia Española de Neurocirugía (AEN)

---

1. Introducción y Contexto

La Academia Española de Neurocirugía (AEN) requiere implementar un sistema automatizado que monitoree las fuentes oficiales donde se publican convocatorias MIR y oposiciones de neurocirugía en España, alertando automáticamente a los usuarios registrados cuando se detecten nuevas plazas.

Este sistema debe funcionar de manera continua, con notificaciones en tiempo real a través de múltiples canales (email, notificaciones web push, SMS opcionales) e integrarse de forma natural en la plataforma web existente de la AEN.

Objetivos Principales

1. Monitoreo automático y continuo de todas las fuentes oficiales
2. Detección inmediata de nuevas convocatorias
3. Notificación a usuarios suscritos en múltiples canales
4. Personalización según especialización y comunidad autónoma
5. Historial y archivo de convocatorias para consulta posterior
6. Bajo costo operativo (preferiblemente gratuito o freemium)

---

2. Fuentes Oficiales a Monitorear

2.1 Convocatorias MIR Nacionales

BOE (Boletín Oficial del Estado)

- **URL Principal:** https://www.boe.es
- **Portal FSE:** https://fse.mscbs.gob.es
- **Ministerio de Sanidad:** https://www.sanidad.gob.es/gabinete/notasPrensa.do
- **Búsqueda específica:** https://www.boe.es/organismo/empleo/
- **Información:** Publicación oficial de todas las convocatorias MIR
- **Frecuencia:** Anual (típicamente agosto para el examen del año siguiente)
- **Especialidad:** Neurocirugía - Orden SND/928/2025 (54 plazas en sector público, 2 en privado)

RSS Disponible:

- Alertas BOE: https://www.boe.es/alertas/alerta.php?tipo=PER
- Requiere registro en "Mi BOE" para recibir alertas personalizadas

Buscador Oficial de Empleo Público

- **URL:** https://administracion.gob.es/pag_Home/empleoPublico/buscador.html
- **Descripción:** Portal centralizado del gobierno español para todas las convocatorias de empleo público
- **Búsqueda:** Filtrar por "Neurocirugía" y tipo de proceso
- **Frecuencia:** Actualización diaria

2.2 Oposiciones Autonómicas por Servicio de Salud

ANDALUCÍA - Servicio Andaluz de Salud (SAS)

Portales oficiales:

- **BOJA (Boletín Oficial Junta Andalucía):** https://www.juntadeandalucia.es/boja
- **Portal RRHH SAS:** https://www.juntadeandalucia.es/organismos/salud
- **Buscar:** "Neurocirugía" + "Facultativo Especialista"

Convocatorias recientes (2026):

- Resolución 6 marzo 2026: Oposición Neurocirugía (concurso-oposición)
- Resolución 3 marzo 2026: Jefe/a de Servicio Neurocirugía - Hospital Virgen del Rocío

Bolsa de empleo temporal:

- URL: https://sanidad.juntadeandalucia.es/

CASTILLA Y LEÓN - SACYL (Servicio de Salud)

Portales oficiales:

- **Portal de Empleo SACYL:** https://empleo.jcyl.es
- **Bolsa de Empleo Abierta:** https://www.saludcastillayleon.es/profesionales/en/bolsa
- **Ofertas y Concursos:** https://www.saludcastillayleon.es/profesionales/en/ofertasconcursos

Especialidades ofertadas:

- Incluye Licenciado/a Especialista y Facultativo/a Especialista en Neurocirugía
- Bolsa abierta y permanente

MADRID - SERMAS (Servicio Madrileño de Salud)

Portales oficiales:

- **Portal Empleo SERMAS:** https://gestiona.madrid.org/bsan_consulta/
- **Bolsas de Contratación:** https://www.comunidad.madrid/servicios/salud/bolsas-contratacion-temporal-servicio-madrileno-salud
- **Bolsa SERMAS:** Acceso a gestión de candidatos

Características:

- Bolsas únicas de empleo temporal
- Proceso de consolidación de empleo temporal

CATALUÑA - ICS (Institut Català de la Salut)

Portales oficiales:

- **Sede PROP Generalitat:** https://sede.gva.es/ (ofertas catalanas publicadas)
- **Portal ICS:** https://www.gencat.cat/salutweb
- **Búsqueda:** "Facultativo/a Especialista" + "Neurocirugía"

Características:

- Bolsas de empleo con baremo de 100 puntos
- Concursos de méritos según Ley 20/2021

CASTILLA-LA MANCHA - SESCAM

Portales oficiales:

- **Portal Empleo SESCAM:** https://sanidad.castillalamancha.es/profesionales/atencion-al-profesional/bolsas-constituidas
- **Bolsas de Empleo:** Procesos selectivos de Neurocirugía
- **Convocatorias:** Calendario de oposiciones

Información reciente:

- Procesos selectivos de Neurocirugía (fecha: 19 de abril 2026)

GALICIA - SERGAS (Servicio Gallego de Salud)

Portales oficiales:

- **Portal SERGAS:** https://www.sergas.es
- **Empleo Público:** Bolsas y oposiciones especializadas
- **Búsqueda:** "Neurocirugía"

PAÍS VASCO - OSAKIDETZA

Portales oficiales:

- **OSAKIDETZA:** https://www.osakidetza.euskadi.eus
- **OPE Estabilización:** https://www.osakidetza.euskadi.eus/ope-estabilizacion/webosk00-procon/es/
- **Empleo público:** Procesos selectivos de especialistas

2.3 Otras Fuentes Complementarias

Portales de Empleo General (con filtro sanitario)

1. **InfoJobs Sanidad:** https://www.infojobs.net (filtrar por "neurocirugía" + "pública")

2. **LinkedIn Jobs:** https://www.linkedin.com/jobs (búsqueda por especialidad)

3. **Bolsas Sanitarias Portales Privados:** Colegios profesionales

Colegios y Asociaciones Profesionales

- **Colegio de Médicos de España:** Publicaciones de oposiciones
- **Asociación Española de Neurocirugía:** Informaciones sobre convocatorias
- **Listas de distribución de residentes:** Información compartida

---

3. Tecnologías y Herramientas de Monitoreo

3.1 Opciones Técnicas (De menor a mayor complejidad)

OPCIÓN 1: Herramientas SaaS de Monitoreo Web (RECOMENDADO PARA INICIO)

Distill.io

- **URL:** https://distill.io/
- **Ventajas:**

- Gratuito hasta 25 monitores (suficiente para fase inicial)
- Monitoreo cada 6 horas en plan gratuito
- Selectores CSS/XPATH personalizados
- Extensión Chrome/Firefox
- Historial de cambios visual
- Integración con Webhooks (Slack, Discord)
- **Limitaciones:**
- Plan gratuito: 6 horas entre chequeos
- Plan de pago desde $5/mes para monitoreo cada 6 minutos
- **Casos de uso:**
- Monitorear BOE.es (sección empleo)
- Monitorear portales autonómicos
- Alertas a Slack/Discord del equipo

Visualping

- **URL:** https://visualping.io/
- **Ventajas:**
- No requiere cuenta previa para crear monitor
- Interfaz muy intuitiva
- Comparativa visual de cambios
- Alternativa premium a Distill.io
- Soporte 24/7
- **Limitaciones:**
- Modelo de pago más caro (desde €9/mes)
- Mejor para monitoreo menos frecuente

OPCIÓN 2: API del BOE + Scripting Custom

BOE API

- **Documentación:** https://www.boe.es/datosabiertos/api/api.php
- **GitHub:** https://github.com/BOE-API/BOE_API
- **Ventajas:**
- API oficial, 100% legal
- Acceso a sumarios del BOE
- Base de datos con ~2,000,000 documentos
- Límite: 1000 elementos por página
- Gratuito
- **Implementación:**
- Endpoint para búsqueda de convocatorias
- Filtrar por "neurocirugía", "oposición", "MIR"
- Polling cada 6-12 horas

Ejemplo de uso:

```
# Pseudocódigo para API BOE
import requests

url = "https://www.boe.es/datosabiertos/api/api.php"
params = {
    'q': 'neurocirugía oposición',
    'resultados_por_pagina': 50,
    'sort': 'fecha:desc'
}

response = requests.get(url, params=params)
nuevas_convocatorias = response.json()
```

Alternativa: GitHub BOE-scraping

- **Repositorio:** https://github.com/Quantika14/BOE-scraping
- **Descripción:** Script Python para descargar BOE
- **Ventajas:** Código abierto, personalizable
- **Desventaja:** Mantenimiento requerido

OPCIÓN 3: Web Scraping Personalizado

Stack Recomendado: Node.js + Express

Librerías:

1. **Cheerio** - Análisis HTML

```
npm install cheerio
// jQuery-like API para parsing
// Ligero y rápido
```

2. **Axios** - Peticiones HTTP

```
npm install axios
// Manejo de requests, timeouts, reintentos
```

3. **Puppeteer** - Para sitios con JavaScript

```
npm install puppeteer
// Navegador headless Chromium
// Necesario para portales autonómicos complejos
```

4. **Cron** - Tareas programadas

```
npm install cron
// Ejecutar scraping cada 6 horas automáticamente
```

5. **Nodemailer** - Envío de emails

```
npm install nodemailer
// O integración con SendGrid/Mailgun
```

Estructura básica:

```
// server.js - Express con monitoreo automático
const express = require('express');
const cheerio = require('cheerio');
const axios = require('axios');
const cron = require('cron');
const nodemailer = require('nodemailer');

const app = express();

// Tarea programada cada 6 horas
const job = cron.schedule('0 */6 * * *', async () => {
  console.log('Iniciando scraping de convocatorias...');
```

```
  try {
    // 1. Scrapear BOE
    const boeConvocatorias = await scrapBOE();

    // 2. Scrapear portales autonómicos
    const autocomConvocatorias = await scrapAutonomicas();

    // 3. Buscar nuevas convocatorias en BD
    const nuevas = await
buscarNuevas([...boeConvocatorias, ...autocomConvocatorias]);

    // 4. Notificar usuarios
    if (nuevas.length > 0) {
      await notificarUsuarios(nuevas);
    }
  } catch (error) {
    console.error('Error en scraping:', error);
  }
});

// Función para scrapear BOE
async function scrapBOE() {
  const url = 'https://www.boe.es/datosabiertos/api/api.php?q=neurocirugía';
  const response = await axios.get(url);
  return response.data.results || [];
}

// API endpoint para consultar convocatorias
app.get('/api/convocatorias', async (req, res) => {
  // Retornar lista de convocatorias almacenadas
});

app.listen(3000, () => {
  console.log('Servidor de alertas escuchando en puerto 3000');
});
```

OPCIÓN 4: Integraciones sin código - Zapier/IFTTT

Zapier

- **URL:** https://zapier.com
- **Ventajas:**
- No requiere código
- Integración con 1000+ servicios
- Manejo de webhooks
- Logs detallados
- **Flujo sugerido:**

1. Monitoreo web con Distill.io

2. Webhook → Zapier

3. Zapier → Envío de email (SendGrid)

4. Zapier → Notificación Slack/Discord

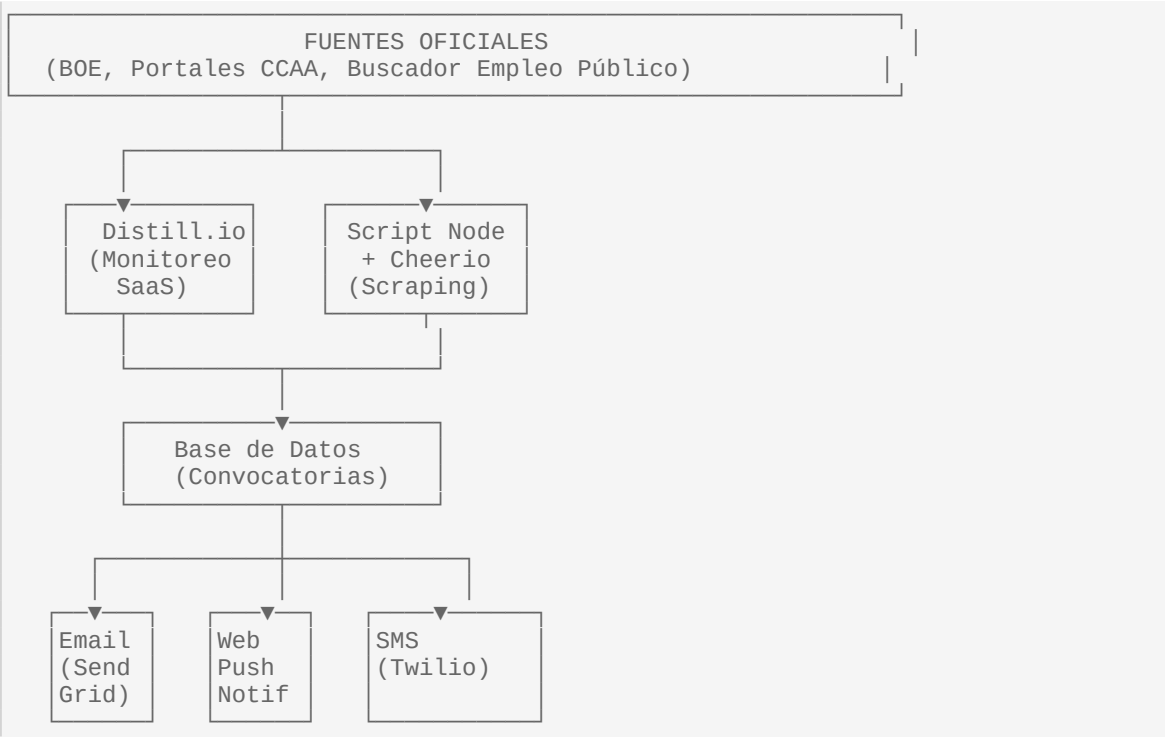
- **Costo:** Desde $25/mes para automatizaciones frecuentes

IFTTT

- **URL:** https://ifttt.com
- **Ventajas:** Enfoque consumer, muy simple
- **Desventaja:** Menos control empresarial

3.2 Recomendación de Arquitectura

Para Academia Española de Neurocirugía: OPCIÓN HÍBRIDA RECOMENDADA



Fase 1 (Mes 1):

- Monitoreo manual con Distill.io de BOE y 3 portales principales
- Alertas por email a grupo administrador

Fase 2 (Mes 2-3):

- Implementar servidor Node.js en VPS/AWS
- Script de scraping para portales autonómicos
- Integración con BD MySQL/PostgreSQL
- Notificaciones a usuarios registrados

Fase 3 (Mes 4+):

- Web push notifications integradas en web
- Dashboard de administración
- Histórico completo de convocatorias
- Filtros personalizados por usuario

4. Servicio de Notificaciones

4.1 Email - SendGrid (Recomendado)

Ventajas:

- 6000 emails gratis durante primeros 30 días
- 100 emails/día indefinidamente en plan gratuito
- API moderna y confiable
- Seguimiento de entregas

- Plantillas HTML

Integración Node.js:

```
const sgMail = require('@sendgrid/mail');
sgMail.setApiKey(process.env.SENDGRID_API_KEY);

async function enviarAlerta(usuario, convocatoria) {
  const msg = {
    to: usuario.email,
    from: 'alertas@aen.es',
    subject: `Nueva convocatoria: ${convocatoria.especialidad}`,
    html: `
      <h2>Nueva Convocatoria de ${convocatoria.especialidad}</h2>
      <p><strong>Especialidad:</strong> ${convocatoria.especialidad}</p>
      <p><strong>Comunidad Autónoma:</strong> ${convocatoria.ccaa}</p>
      <p><strong>Plazas:</strong> ${convocatoria.plazas}</p>
      <p><strong>Fuente:</strong> <a href="${convocatoria.url}">$
{convocatoria.fuente}</a></p>
      <p><strong>Fecha de publicación:</strong> ${convocatoria.fecha}</p>
      <a href="https://aen.es/convocatorias/${convocatoria.id}" class="btn">Ver
detalles</a>
    `,
  };

  await sgMail.send(msg);
}
```

Alternativa: Mailgun (muy similar, 5000 emails/mes gratis)

4.2 Web Push Notifications

Implementación con Service Workers:

```
// public/service-worker.js
self.addEventListener('push', event => {
  const data = event.data.json();

  const options = {
    body: `${data.ccaa}: ${data.plazas} plazas en ${data.especialidad}`,
    icon: '/images/logo.png',
    badge: '/images/badge.png',
    tag: 'convocatoria-' + data.id,
    requireInteraction: true, // Usuario debe interactuar
    actions: [
      {
        action: 'open',
        title: 'Ver convocatoria'
      },
      {
        action: 'close',
        title: 'Cerrar'
      }
    ]
  };

  event.waitUntil(
    self.registration.showNotification(`Nueva convocatoria: ${data.especialidad}
`, options)
  );
});
```

```
// JavaScript en la página
if ('serviceWorker' in navigator && 'PushManager' in window) {
  navigator.serviceWorker.register('/service-worker.js');

  // Solicitar permiso al usuario
  Notification.requestPermission().then(permission => {
    if (permission === 'granted') {
      // Suscribir a push notifications
      navigator.serviceWorker.ready.then(registration => {
        registration.pushManager.subscribe({
          userVisibleOnly: true,
          applicationServerKey: urlBase64ToUint8Array(PUBLIC_VAPID_KEY)
        });
      });
    }
  });
}
```

Ventajas:

- Funciona incluso si el navegador está cerrado
- Visibilidad máxima
- Sin dependencia externa

4.3 SMS (Opcional - Twilio)

Para alertas críticas (nuevas plazas de neurocirugía MIR):

```
const twilio = require('twilio');
const client = twilio(process.env.TWILIO_ACCOUNT_SID,
process.env.TWILIO_AUTH_TOKEN);

async function enviarAlertaSMS(telefono, convocatoria) {
  await client.messages.create({
    body: `ALERTA AEN: ${convocatoria.plazas} plazas ${convocatoria.ccaa} - $
{convocatoria.especialidad}. Ver: aen.es/conv/${convocatoria.id}`,
    from: process.env.TWILIO_PHONE_NUMBER,
    to: telefono
  });
}
```

Costo: $0.0075 por SMS (bajo volumen)

---

5. Integración en la Web HTML Existente

5.1 Formulario de Suscripción

```
<!-- En la página principal o sección de alertas -->
<section id="alertas">
  <h2>Suscribirse a Alertas de Convocatorias</h2>

  <form id="formularioAlertaS" method="POST" action="/api/suscribir">
    <div class="form-group">
      <label for="email">Email:</label>
      <input type="email" id="email" name="email" required>
    </div>

    <div class="form-group">
```

```
      <label for="telefono">Teléfono (opcional para SMS):</label>
      <input type="tel" id="telefono" name="telefono">
    </div>

    <div class="form-group">
      <label>Especialidades de interés:</label>
      <label>
        <input type="checkbox" name="especialidades" value="neurocirugía"
checked>
        Neurocirugía General
      </label>
      <label>
        <input type="checkbox" name="especialidades" value="oncología-
neuroquirurgica">
        Oncología Neuroquirúrgica
      </label>
      <label>
        <input type="checkbox" name="especialidades" value="cirugía-medular">
        Cirugía Medular
      </label>
    </div>

    <div class="form-group">
      <label>Comunidades Autónomas de interés:</label>
      <select name="ccaa" multiple>
        <option value="todas" selected>Todas</option>
        <option value="andalucia">Andalucía</option>
        <option value="madrid">Madrid</option>
        <option value="cataluna">Cataluña</option>
        <!-- más opciones -->
      </select>
      <small>(Dejar vacío para recibir todas)</small>
    </div>

    <div class="form-group">
      <label>Canales de notificación:</label>
      <label>
        <input type="checkbox" name="canales" value="email" checked>
        Email
      </label>
      <label>
        <input type="checkbox" name="canales" value="web-push">
        Notificaciones Web
      </label>
      <label>
        <input type="checkbox" name="canales" value="sms">
        SMS
      </label>
    </div>

    <button type="submit" class="btn-primary">Suscribirse</button>
  </form>

  <div id="mensaje-respuesta" class="alert" style="display:none;"></div>
</section>

<script>
// Habilitar web push notifications si el usuario lo selecciona
document.getElementById('formularioAlertaS').addEventListener('change',
function(e) {
  if (e.target.name === 'canales' && e.target.value === 'web-push' &&
e.target.checked) {
    if ('serviceWorker' in navigator && 'PushManager' in window) {
      Notification.requestPermission();
    }
  }
```

```
});

// Envío del formulario
document.getElementById('formularioAlertaS').addEventListener('submit', async
(e) => {
  e.preventDefault();

  const formData = new FormData(e.target);
  const data = {
    email: formData.get('email'),
    telefono: formData.get('telefono'),
    especialidades: formData.getAll('especialidades'),
    ccaa: formData.getAll('ccaa'),
    canales: formData.getAll('canales')
  };

  try {
    const response = await fetch('/api/suscribir', {
      method: 'POST',
      headers: { 'Content-Type': 'application/json' },
      body: JSON.stringify(data)
    });

    const resultado = await response.json();

    if (response.ok) {
      document.getElementById('mensaje-respuesta').textContent =
        'Te has suscrito correctamente. Recibirás alertas en ' +
data.canales.join(', ');
      document.getElementById('mensaje-respuesta').className = 'alert alert-
success';
    }
  } catch (error) {
    console.error('Error:', error);
  }
});
</script>
```

5.2 Dashboard de Convocatorias

```
<section id="convocatorias">
  <h2>Convocatorias Activas y Próximas</h2>

  <!-- Filtros -->
  <div class="filtros">
    <input type="text" id="filtro-especialidad" placeholder="Filtrar por
especialidad...">
    <select id="filtro-ccaa">
      <option value="">Todas las CCAA</option>
      <option value="andalucia">Andalucía</option>
      <option value="madrid">Madrid</option>
      <!-- más opciones -->
    </select>
    <select id="filtro-tipo">
      <option value="">Todos los tipos</option>
      <option value="mir">MIR</option>
      <option value="oposicion">Oposición</option>
      <option value="bolsa">Bolsa de Empleo</option>
    </select>
  </div>

  <!-- Tabla de convocatorias -->
```

```
  <div id="contenedor-convocatorias">
    <table class="tabla-convocatorias">
      <thead>
        <tr>
          <th>Especialidad</th>
          <th>CCAA</th>
          <th>Tipo</th>
          <th>Plazas</th>
          <th>Plazos</th>
          <th>Enlace</th>
          <th>Acción</th>
        </tr>
      </thead>
      <tbody id="tabla-body">
        <!-- Se llena dinámicamente con JavaScript -->
      </tbody>
    </table>
  </div>
</section>

<script>
// Cargar convocatorias desde API
async function cargarConvocatorias(filtros = {}) {
  try {
    const params = new URLSearchParams(filtros);
    const response = await fetch(`/api/convocatorias?${params}`);
    const convocatorias = await response.json();

    const tbody = document.getElementById('tabla-body');
    tbody.innerHTML = '';

    convocatorias.forEach(conv => {
      const row = `
        <tr>
          <td>${conv.especialidad}</td>
          <td>${conv.ccaa}</td>
          <td><span class="badge-${conv.tipo}">${conv.tipo}</span></td>
          <td>${conv.plazas}</td>
          <td>
            <small>Solicitud: ${new
Date(conv.fecha_solicitud).toLocaleDateString('es-ES')}</small><br>
            <small>Examen: ${new Date(conv.fecha_examen).toLocaleDateString('es-
ES')}</small>
          </td>
          <td><a href="${conv.url}" target="_blank">Ver BOE/Portal</a></td>
          <td>
            <button onclick="guardarConvocatoria('${conv.id}')">Guardar</button>
            <button onclick="compartir('${conv.id}')">Compartir</button>
          </td>
        </tr>
      `;
      tbody.innerHTML += row;
    });
  } catch (error) {
    console.error('Error cargando convocatorias:', error);
  }
}

// Aplicar filtros
document.getElementById('filtro-especialidad').addEventListener('change', () =>
{
  const filtros = {
    especialidad: document.getElementById('filtro-especialidad').value,
    ccaa: document.getElementById('filtro-ccaa').value,
    tipo: document.getElementById('filtro-tipo').value
  };
  cargarConvocatorias(filtros);
```

```
});

// Cargar inicialmente
cargarConvocatorias();
</script>
```

5.3 Página de Historial y Estadísticas

```
<section id="estadisticas">
  <h2>Historial de Convocatorias y Estadísticas</h2>

  <div class="stats-grid">
    <div class="stat-card">
      <h3 id="total-convocatorias">--</h3>
      <p>Convocatorias Totales (2024-2026)</p>
    </div>
    <div class="stat-card">
      <h3 id="total-plazas">--</h3>
      <p>Plazas Ofertadas (Neurocirugía)</p>
    </div>
    <div class="stat-card">
      <h3 id="regiones-activas">--</h3>
      <p>Comunidades con Convocatorias</p>
    </div>
    <div class="stat-card">
      <h3 id="proxima-convocatoria">--</h3>
      <p>Próxima Convocatoria MIR</p>
    </div>
  </div>

  <!-- Tabla histórica -->
  <h3>Archivo Completo</h3>
  <table class="tabla-historico">
    <thead>
      <tr>
        <th>Año</th>
        <th>Tipo</th>
        <th>Especialidad</th>
        <th>Plazas</th>
        <th>Detalles</th>
      </tr>
    </thead>
    <tbody id="historico-body">
      <!-- Se llena dinámicamente -->
    </tbody>
  </table>
</section>
```

6. Arquitectura Técnica Detallada

6.1 Stack Tecnológico Recomendado

```
Frontend:
- HTML5, CSS3, JavaScript vanilla (sin dependencias pesadas)
- Service Workers para push notifications
- Fetch API para comunicación con backend

Backend:
```

```
- Node.js v18+
- Express.js para servidor web
- Cheerio para web scraping
- Axios para peticiones HTTP
- Cron para tareas programadas

Almacenamiento:
- PostgreSQL (convocatorias, usuarios)
- Redis (caché de alertas, sesiones)

Infraestructura:
- Docker para contenedores
- AWS EC2 / DigitalOcean (servidor VPS)
- GitHub para control de versiones

Notificaciones:
- SendGrid (email)
- Web Push Notifications (navegador)
- Twilio (SMS opcional)
```

6.2 Esquema de Base de Datos

```
-- Tabla de usuarios suscritos
CREATE TABLE usuarios (
  id INT PRIMARY KEY AUTO_INCREMENT,
  email VARCHAR(255) UNIQUE NOT NULL,
  telefono VARCHAR(20),
  fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  activo BOOLEAN DEFAULT TRUE,
  suscrito_email BOOLEAN DEFAULT TRUE,
  suscrito_webpush BOOLEAN DEFAULT FALSE,
  suscrito_sms BOOLEAN DEFAULT FALSE,
  preferencias_json JSON
);

-- Tabla de convocatorias
CREATE TABLE convocatorias (
  id INT PRIMARY KEY AUTO_INCREMENT,
  titulo VARCHAR(255) NOT NULL,
  especialidad VARCHAR(100) NOT NULL,
  ccaa VARCHAR(100),
  tipo_proceso ENUM('mir', 'oposicion', 'bolsa', 'otro'),
  plazas INT,
  fuente VARCHAR(50), -- 'BOE', 'SAS', 'SERMAS', etc.
  url TEXT UNIQUE,
  fecha_publicacion TIMESTAMP,
  fecha_solicitud_inicio DATE,
  fecha_solicitud_fin DATE,
  fecha_examen DATE,
  contenido_html LONGTEXT,
  hash_contenido VARCHAR(64), -- Para detectar cambios
  fecha_deteccion TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  notificada BOOLEAN DEFAULT FALSE
);

-- Tabla de notificaciones enviadas
CREATE TABLE notificaciones (
  id INT PRIMARY KEY AUTO_INCREMENT,
  convocatoria_id INT,
  usuario_id INT,
  canal ENUM('email', 'webpush', 'sms'),
  fecha_envio TIMESTAMP,
  estado ENUM('enviada', 'pendiente', 'error'),
```

```
  detalles_error TEXT,
  FOREIGN KEY (convocatoria_id) REFERENCES convocatorias(id),
  FOREIGN KEY (usuario_id) REFERENCES usuarios(id)
);

-- Tabla de preferencias por usuario
CREATE TABLE preferencias_usuario (
  id INT PRIMARY KEY AUTO_INCREMENT,
  usuario_id INT,
  especialidad VARCHAR(100),
  ccaa VARCHAR(100),
  tipo_proceso VARCHAR(50),
  FOREIGN KEY (usuario_id) REFERENCES usuarios(id)
);
```

6.3 Endpoints API

```
POST /api/suscribir
  - Registrar nuevo usuario
  - Body: { email, telefono?, especialidades[], ccaa[], canales[] }

GET /api/convocatorias
  - Obtener listado de convocatorias
  - Query params: especialidad?, ccaa?, tipo?, page?, limit?
  - Response: { total, pagina, resultados: [] }

GET /api/convocatorias/:id
  - Obtener detalles de una convocatoria
  - Response: { id, titulo, especialidad, ... }

POST /api/usuarios/:id/guardar-convocatoria/:convId
  - Marcar convocatoria como guardada para el usuario

POST /api/usuarios/:id/preferencias
  - Actualizar preferencias del usuario
  - Body: { especialidades[], ccaa[], canales[] }

GET /api/estadisticas
  - Estadísticas de convocatorias
  - Response: { totalConvocatorias, totalPlazas, distribucionCCAA, ... }

POST /api/desuscribir
  - Desuscribirse de alertas
  - Body: { email }
```

7. Plan de Implementación por Fases

Fase 1: MVP (Semanas 1-4)

Objetivo: Sistema funcional con monitoreo básico y notificaciones por email

Tareas:

1. **Infraestructura básica** (Semana 1)

- Registrar en SendGrid (gratuito)
- Crear servidor VPS ($5/mes DigitalOcean)
- Configurar dominio de email (alertas@aen.es)

- Inicializar repositorio Git

2. **Monitoreo inicial** (Semana 2)
   - Registrar en Distill.io
   - Crear monitores para:
   - BOE.es (sección empleo)
   - administracion.gob.es (buscador)
   - juntadeandalucia.es/boja (SAS)
   - Configurar alertas a email administrativo
3. **Backend básico** (Semana 3)
   - Instalar Node.js + Express
   - Crear BD PostgreSQL
   - Implementar endpoint POST /api/suscribir
   - Crear script para guardar convocatorias en BD
4. **Frontend suscripción** (Semana 4)
   - Crear formulario de suscripción en HTML
   - Validación cliente-lado
   - Integración con API backend
   - Deploy en servidor

Entregables:

- ✓ Sistema recibiendo alertas de Distill.io
- ✓ Emails siendo enviados con SendGrid
- ✓ BD almacenando convocatorias
- ✓ Al menos 10 usuarios suscritos

Fase 2: Expansión de Fuentes (Semanas 5-8)

Objetivo: Monitoreo de todos los portales autonómicos

Tareas:

1. **Web scraping personalizado** (Semana 5-6)
   - Crear script Node.js con Cheerio
   - Scrapear BOE API oficial
   - Scrapear 6 portales CCAA principales (SAS, SERMAS, SACYL, ICS, SERGAS, OSAKIDETZA)
   - Implementar caché para no duplicar
2. **Automatización** (Semana 7)
   - Configurar cron para ejecutar scraping cada 6 horas
   - Detectar nuevas convocatorias automáticamente
   - Enviar emails a usuarios suscritos
3. **Dashboard** (Semana 8)
   - Crear página con tabla de convocatorias
   - Filtros por especialidad, CCAA, tipo
   - Enlace a fuentes originales
   - Sistema de búsqueda

Entregables:

- ✓ 50+ convocatorias en BD
- ✓ Monitoreo automatizado 24/7
- ✓ Dashboard funcional
- ✓ 100+ usuarios suscritos

Fase 3: Notificaciones Avanzadas (Semanas 9-12)

Objetivo: Web push notifications y SMS

Tareas:

1. **Web Push Notifications** (Semana 9-10)
   - Implementar Service Worker
   - Solicitar permisos a usuarios
   - Enviar notificaciones en tiempo real
   - Historial de notificaciones
2. **SMS (Opcional)** (Semana 11)
   - Integración con Twilio
   - Para alertas críticas de MIR
   - Confirmar suscripción
3. **Historial y estadísticas** (Semana 12)
   - Página de archivo histórico
   - Estadísticas (plazas por año, distribución CCAA)
   - Exportar a CSV/PDF

Entregables:

- ✓ Web push notifications funcionando
- ✓ SMS para usuarios premium
- ✓ Historial completo desde 2020
- ✓ Dashboard de estadísticas

---

8. Operación y Mantenimiento

8.1 Monitoreo del Sistema

```
# Verificar estado del servidor cada 5 minutos
*/5 * * * * curl -s https://aen.es/api/health | grep -q "ok" || alert_admin

# Logs de scraping
tail -f /var/log/aen-scraper.log

# Verificar BD
SELECT COUNT(*) FROM convocatorias WHERE fecha_deteccion > DATE_SUB(NOW(),
INTERVAL 24 HOUR);

# Estado de emails
SELECT COUNT(*) FROM notificaciones WHERE estado = 'error' AND fecha_envio >
DATE_SUB(NOW(), INTERVAL 24 HOUR);
```

8.2 Frecuencia de Monitoreo

```
- BOE.es: Cada 6 horas (máximo cambio es diario)
- Portales CCAA: Cada 6 horas (variable por región)
- Buscador Empleo Público: Cada 12 horas
- Verificación de emails: Cada hora (máximo 100 intentos)
```

8.3 Actualización de Fuentes

Revisar mensualmente:

- URLs de portales (pueden cambiar URLs)
- Estructura HTML (afecta al scraping)
- Nuevos portales autonómicos
- Cambios en BOE API

8.4 Gestión de Suscriptores

```
-- Eliminar emails inválidos (bounce)
DELETE FROM usuarios WHERE bounces > 5;

-- Inactivar usuarios sin clicks en 6 meses
UPDATE usuarios SET activo = FALSE
WHERE id NOT IN (
  SELECT DISTINCT usuario_id FROM notificaciones
  WHERE fecha_envio > DATE_SUB(NOW(), INTERVAL 6 MONTH)
);

-- Enviar emails de confirmación trimestrales
UPDATE usuarios SET
  confirmacion_requerida = TRUE
WHERE activo = TRUE AND
  ultima_confirmacion < DATE_SUB(NOW(), INTERVAL 90 DAY);
```

---

9. Consideraciones Legales y Éticas

9.1 Conformidad Legal

- **Scraping**: Respetar robots.txt de cada portal
- **RGPD**: Consentimiento explícito, derecho al olvido, política de privacidad
- **Bots**: No sobrecargar servidores (max 1 request cada 5 minutos por fuente)

9.2 Política de Privacidad

Elemento obligatorio:

```
Los datos personales (email, teléfono) son protegidos según RGPD.
No se venden ni comparten con terceros.
El usuario puede desuscribirse en cualquier momento.
Se guardan registros de notificaciones por máximo 12 meses.
```

9.3 Términos de Servicio

- Sistema informativo, no reemplaza consultas oficiales
- Alertas "best effort", no garantía de 100% cobertura
- AEN no es responsable de errores en fuentes oficiales

---

10. Presupuesto y Costos

10.1 Costos Iniciales

| Concepto | Costo | Proveedor |

|----------|-------|-----------|

| Dominio (alertas.aen.es) | €10/año | Registro.es |

| Certificado SSL | Gratuito | Let's Encrypt |

| Servidor VPS | €5/mes | DigitalOcean |

| Base de datos | Incluido | DigitalOcean |

| **Total Inicial** | **€70** | |

10.2 Costos Mensuales Recurrentes

| Concepto | Costo | Uso |

|----------|-------|-----|

| Servidor VPS | €5 | 1 CPU, 1GB RAM (scalable) |

| SendGrid | Gratuito | Hasta 100 emails/día |

| Distill.io | Gratuito | 25 monitores máximo |

| Twilio SMS | $0.01 c/u | Solo si se habilita |

| CDN (opcional) | €0-10 | Caché de assets |

| **Total Mensual** | **€5-15** | |

10.3 Escenarios de Crecimiento

- **1000 usuarios**: €15/mes (SendGrid aún gratuito)
- **5000 usuarios**: €30/mes (SendGrid $25 + servidor upgrade)
- **10000+ usuarios**: €50-100/mes (infraestructura escalada)

---

11. Alternativas y Comparativa

11.1 Soluciones Existentes

Opobusca (Competidor)

- URL: https://www.opobusca.com
- **Ventaja:** Base de datos grande, interfaz amigable
- **Desventaja:** No es especializado en neurocirugía, modelo de negocio unclear

Portales Autonómicos Nativos

- **Ventaja:** Información oficial, confiable
- **Desventaja:** Fragmentados, difícil monitoreo centralizado, sin notificaciones

Grupo CTO / ProMir (Preparadores)

- URL: https://grupocto.es, https://promir.es
- **Ventaja:** Especializado en MIR, interfaz premium
- **Desventaja:** Enfocado en preparación, no alertas tiempo real

11.2 Por qué no usar soluciones existentes

1. **Especialización:** AEN puede ofrecer alertas 100% especializadas en neurocirugía

2. **Valor agregado:** Diferenciador competitivo para curso

3. **Control:** Datos propios, sin dependencia de terceros

4. **Costo:** Mucho más económico que soluciones premium

5. **Integración:** Perfectamente integrado en plataforma AEN

---

12. Próximos Pasos

Inmediatos (Esta semana):

1. **Aprobación del proyecto**
   - [ ] Revisión de este documento
   - [ ] Presupuesto aprobado
   - [ ] Equipo asignado
2. **Preparación técnica**
   - [ ] Registrar dominio (alertas.aen.es)
   - [ ] Registrar en SendGrid
   - [ ] Crear cuenta DigitalOcean VPS
   - [ ] Crear repositorio GitHub privado
3. **Configuración inicial**
   - [ ] Registrar en Distill.io
   - [ ] Crear 3-4 monitores piloto
   - [ ] Configurar alertas iniciales

Corto Plazo (Semana 1-4):

- [ ] Completar Fase 1 (MVP)
- [ ] Deploy en producción
- [ ] Beta testing con 10-20 usuarios
- [ ] Ajustes iniciales

Mediano Plazo (Semana 5-12):

- [ ] Completar Fases 2 y 3
- [ ] Lanzamiento completo al público
- [ ] Marketing en redes de AEN
- [ ] Optimización basada en feedback

---

13. Referencias y Enlaces Útiles

Portales Oficiales

- **BOE (Boletín Oficial Estado):** https://www.boe.es
- **Ministerio de Sanidad FSE:** https://fse.mscbs.gob.es
- **Buscador Empleo Público:** https://administracion.gob.es/pag_Home/empleoPublico/buscador.html

Tecnologías Recomendadas

- **Node.js:** https://nodejs.org
- **Express.js:** https://expressjs.com
- **Cheerio:** https://cheerio.js.org
- **Distill.io:** https://distill.io
- **SendGrid:** https://sendgrid.com

Documentación Técnica

- **MDN Web Push Notifications:** https://developer.mozilla.org/es/docs/Web/API/Notifications_API/Using_the_Notifications_API
- **BOE API:** https://www.boe.es/datosabiertos/api/api.php
- **Web Scraping ético:** https://realphython.com/beautiful-soup-web-scraper/ (buenas prácticas)

Conclusión

Este sistema de alertas proporcionará a la Academia Española de Neurocirugía una herramienta diferenciadora y de alto valor para sus usuarios residentes en España y Latinoamérica. El enfoque de implementación por fases permite comenzar con un MVP funcional en 4 semanas, escalando gradualmente hacia un sistema completo y robusto.

La inversión inicial es mínima (~€70) y los costos operativos muy bajos (~€5-15/mes), proporcionando un ROI excelente en términos de valor para los usuarios y diferenciación competitiva de la AEN.

Documento preparado: 29 de Marzo de 2026

Versión: 1.0

Estado: Listo para aprobación ejecutiva