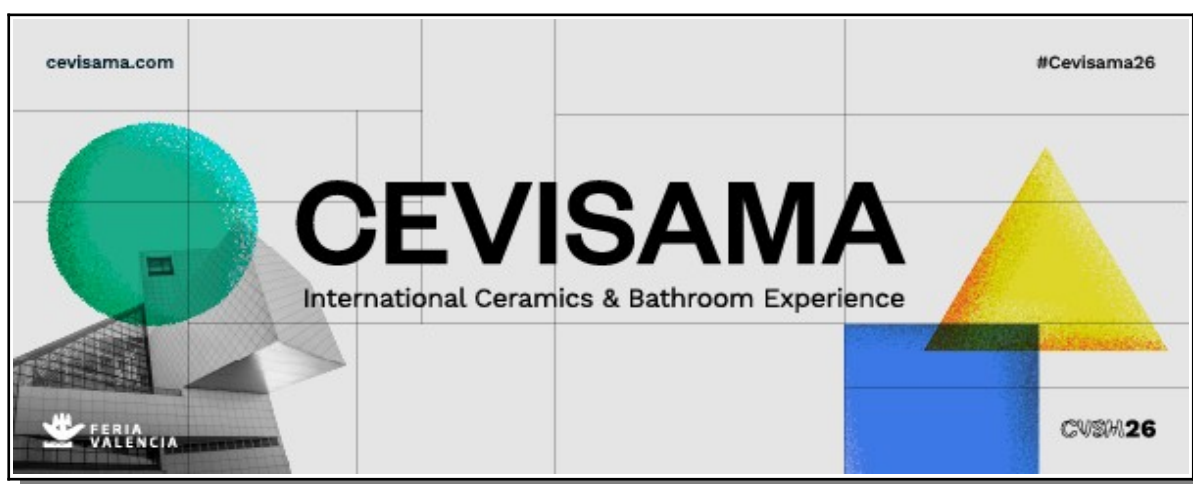


Supuesto 2 Feria Valencia Memoria Cevisama



Índice

1. ANÁLISIS Y PLANIFICACIÓN
2. DISEÑO
3. TEST CON USUARIOS
4. DESARROLLO Y ENVÍO DE CORREOS
5. CONTROL DE VERSIONES CON GITHUB
6. DATOS
7. PERSONALIZACIÓN DEL CORREO MJML
8. VERIFICAR COMPATIBILIDAD CON DIFERENTES CLIENTES DE CORREO

1. ANÁLISIS Y PLANIFICACIÓN

Análisis de Requerimientos

El público objetivo principal de esta campaña son los **expositores actuales y potenciales** de Cevisama 2025. El objetivo es comunicarles la inversión sin precedentes realizada por Feria Valencia para atraer a compradores y prescriptores internacionales de alto nivel, incentivando su participación y maximizando el retorno de su inversión.

DAFO y Competencia

Fortalezas

- **Inversión destacada:** El anuncio de una inversión histórica para Cevisama 2025 es un gran atractivo para los expositores, generando confianza.
- **Marca consolidada:** Cevisama ya es un referente en el sector cerámico, lo que facilita la atención de posibles expositores.
- **Programa VIP reforzado:** La inversión en traer compradores y prescriptores de alto nivel es un beneficio directo para los expositores.
- **Mercados estratégicos:** La campaña se dirige a mercados internacionales clave, aumentando el potencial de negocio.

Debilidades

- **Posible escepticismo:** A pesar de la inversión, algunos expositores pueden dudar hasta ver resultados concretos en la feria.
- **Alta competencia:** Destacar en un buzón de entrada saturado de correos es un reto.
- **Personalización a gran escala:** Garantizar una personalización efectiva para todos los expositores puede ser complejo.

Oportunidades

- **Posicionamiento de liderazgo:** La campaña puede consolidar a Cevisama como líder del sector.
- **Nuevos expositores:** La promesa de mayor retorno de inversión puede atraer a empresas que nunca han participado.
- **Contenido de valor:** La inversión en la feria ofrece material excelente para crear contenido atractivo.

Amenazas

- **Competencia directa:** Otras ferias del sector podrían lanzar campañas similares o más agresivas.

- **Fatiga publicitaria:** Los expositores pueden ignorar correos si perciben demasiada publicidad o poca relevancia.
- **Recesión económica:** Una desaceleración global podría afectar la inversión de las empresas en ferias.

Objetivos de la Campaña

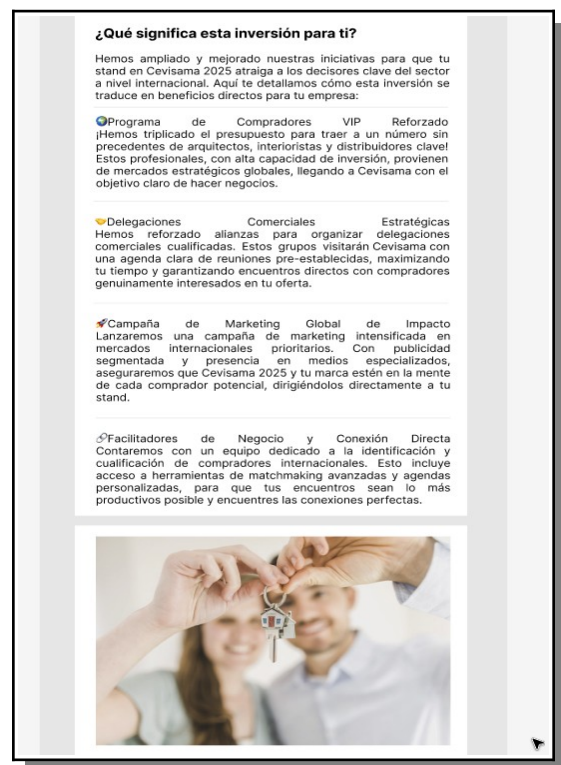
La campaña busca lograr los siguientes objetivos:

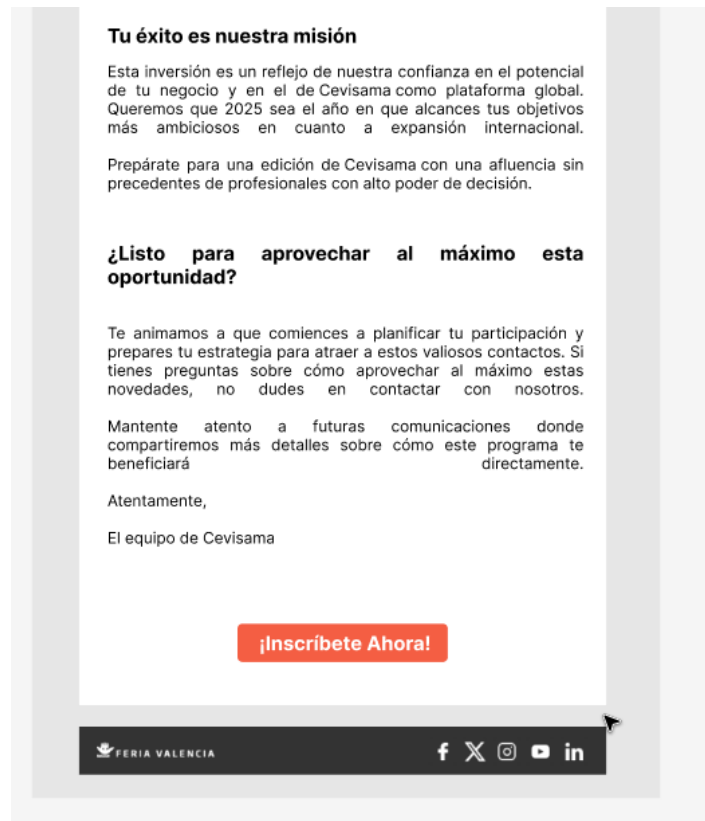
- **Aumentar las inscripciones** y confirmaciones de expositores para Cevisama 2025.
- **Reforzar la imagen de Cevisama** como una plataforma global líder en el sector.
- **Informar sobre las iniciativas** para atraer compradores internacionales.
- **Generar confianza y compromiso** entre los expositores y Feria Valencia.

2. DISEÑO

- Mockup (Figma):

Hemos hecho el diseño en la plataforma online FIGMA:





[Enlace al proyecto de Figma](#)

Representación visual previa.

o Justificación del diseño: para quién, por qué, qué elementos visuales, qué jerarquía.

- Responsive avanzado con MJML:

o Usar media queries (3 media queries) para personalizar según el dispositivo, de forma que el diseño cambie o se oculten elementos según el tamaño de la pantalla.

3. TEST CON USUARIOS

- Selección de muestra: al menos 10 personas reales (profesores, compañeros, familiares).
- Formulario de valoración: Google Forms
- Feedback: Recogida y análisis del resultado de la encuesta
- Rediseño: si es necesario, a partir de los resultados obtenidos.

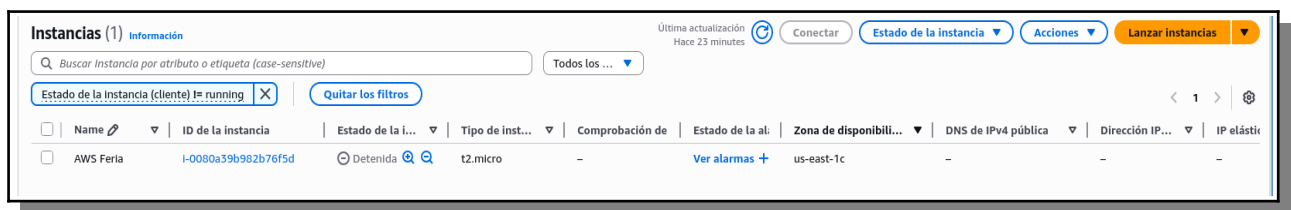
He realizado un formulario de preguntas en Google Forms, en el cuál los usuarios que han recibido la newsletter han dado su opinión.

[Enlace al formulario](#)

4. DESARROLLO Y ENVÍO DE CORREOS

- AWS EC2 + mjml + node.js (javascript):

Hemos configurado una instancia ec2 en AWS Academy con un Ubuntu Server.



- o Lectura de datos de clientes.
- o Automatización del envío con Gmail o O365

Configuración del Sistema de Envío Automatizado de Correos

1. Entorno en AWS EC2:

- Hemos preparado una instancia **EC2** como el servidor donde reside la aplicación.
- Instalado **Node.js** y **npm** para ejecutar un código JavaScript.
- Configurado **PM2** para gestionar y mantener la aplicación Node.js funcionando 24/7 de forma robusta.

2. Base de Datos MySQL:

- Creamos una base de datos (clientesEmail_db) en **MySQL** para almacenar:

- clientes: Información de tus destinatarios (nombre, email).
 - emails_plantillas: Donde se guardan diferentes plantillas MJML.
 - emails_enviados: Para registrar los correos enviados y evitar duplicados.
- Establecida la **conexión** entre tu aplicación Node.js y la base de datos.
3. **Gestión de Variables de Entorno:**
- Creado un archivo .env para almacenar de forma segura credenciales (DB, email) y configuraciones.
 - Implementado la librería **dotenv** en el index.js para cargar estas variables al inicio de tu aplicación.
4. **Lógica de Envío de Correos (Node.js con MJML y Nodemailer):**
- Desarrollado la función sendEmail que:
 - Utiliza **Nodemailer** para conectarse a **Gmail** (con Contraseña de Aplicación para 2FA) o **Outlook 365** (vía SMTP/OAuth).
 - Procesa plantillas de correo en **MJML** (cargadas desde archivos como test_email.mjml en la carpeta templates/).
 - Convierte el MJML a HTML responsivo (mjml2html).
 - Permite la **personalización** de plantillas con datos de clientes (ej. {{nombre}}).
5. **Automatización del Proceso:**
- Implementada la función sendEmailsToClientsFromDB para:
 - Leer datos de clientes desde la base de datos MySQL.
 - Iterar sobre ellos y llamar a sendEmail para cada uno.
 - (Pendiente de implementar completamente) Una lógica para registrar envíos en emails_enviados y evitar reenvíos.
 - Configurado **node-cron** en index.js para **programar la ejecución automática** de sendEmailsToClientsFromDB en intervalos definidos (ej. diariamente a las 9 AM).

```

root@ip-172-31-94-141:/home/ubuntu/servicioEmail# node index.js
Iniciando proceso de envío de correos a clientes desde la base de datos...
Plantilla de cliente cargada desde archivo.
Enviando correo a: [REDACTED] (Rubén [REDACTED])
Correo enviado a [REDACTED] <[REDACTED]@gmail.com>
Enviando correo a: [REDACTED] (Floy)
Correo enviado a [REDACTED] <[REDACTED]@gmail.com>
Enviando correo a: [REDACTED] (Pau)
Correo enviado a [REDACTED] <[REDACTED]@gmail.com>
Enviando correo a: [REDACTED] (Rober)
Correo enviado a [REDACTED] <[REDACTED]@gmail.com>
Enviando correo a: [REDACTED] (Moha)
Correo enviado a [REDACTED] <[REDACTED]@gmail.com>
Enviando correo a: [REDACTED] (Dani)
Correo enviado a [REDACTED] <[REDACTED]@gmail.com>
Enviando correo a: [REDACTED] (Juan)
Correo enviado a [REDACTED] <[REDACTED]@gmail.com>
Enviando correo a: [REDACTED] (Álvaro)
Correo enviado a [REDACTED] <[REDACTED]@gmail.com>
Enviando correo a: [REDACTED] (Irene)
Correo enviado a [REDACTED] <[REDACTED]@gmail.com>
Proceso de envío de correos a clientes finalizado.
root@ip-172-31-94-141:/home/ubuntu/servicioEmail#

```

5. CONTROL DE VERSIONES CON GITHUB

- Repositorio público.
- Extensión integrada en Visual Studio Code.
- Commits regulares + documentación de cambios.
- Uso de ramas

Hemos creado un repositorio público en GitHub en el cual vamos guardando todo el trabajo y código que vamos haciendo. Realizando un commit cuándo creamos algo nuevo o modificamos alguna parte.

Enlace al repositorio: <https://github.com/Mohamed2651/FeriaValencia>

6. DATOS

He creado una base de datos con los usuarios y sus correos a partir de la cuál vamos a acceder a ella para enviar a todos los usuarios la newsletter.


```
mysql> select * from clientes;
```

id	nombre	email	fecha_registro
1	Rubén Ramírez	[REDACTED]	2025-05-27 13:30:35
2	Eloy	[REDACTED]@gmail.com	2025-05-28 13:44:24
3	Pau	[REDACTED]@gmail.com	2025-05-28 13:45:03
4	Rober	[REDACTED]@gmail.com	2025-05-28 13:45:35
5	Moha	[REDACTED]@gva.es	2025-05-28 13:57:54
6	Dani	[REDACTED]@gmail.com	2025-05-28 14:46:48
7	Juan	[REDACTED]@gmail.com	2025-05-28 14:47:39
8	Álvaro	[REDACTED]@gmail.com	2025-05-28 14:48:26
9	Irene	[REDACTED]@gmail.com	2025-05-28 14:49:03

También he configurado un servicio s3 de AWS que es un almacenamiento en la nube, donde guardaremos las imágenes y el output html.

imagenes-publicas-mjml > CEVISAMA/

CEVISAMA/ Copiar URI de S3

Objetos (3) Copiar URI de S3 Copiar URL Descargar Abrir Eliminar Acciones Crear carpeta Cargar

Los objetos son las entidades fundamentales que se almacenan en Amazon S3. Puede utilizar el [Inventario de Amazon S3](#) para obtener una lista de todos los objetos de su bucket. Para que otras personas obtengan acceso a sus objetos, tendrá que concederles permisos de forma explícita. [Más información](#)

Buscar objetos por prefijo Mostrar versiones

<input type="checkbox"/>	Nombre	Tipo	Última modificación	Tamaño	Clase de almacenamiento
<input type="checkbox"/>	Cabecera-CEVISAMA.jpg	Jpg	29 May 2025 6:06:11 PM CEST	60.7 KB	Estándar
<input type="checkbox"/>	img1.jpeg	Jpeg	29 May 2025 6:10:36 PM CEST	1.0 MB	Estándar
<input type="checkbox"/>	img2.jpeg	Jpeg	29 May 2025 6:10:34 PM CEST	521.5 KB	Estándar

7. PERSONALIZACIÓN DEL CORREO MJML

uso de variables (nombre, empresa, idioma, etc.).

Permite la **personalización** de plantillas con datos de clientes ({{nombre}}).

evaluar el impacto de correos personalizados vs genéricos en el test con usuarios

```
<h2>Hola, {{nombre_destinatario}}</h2>
```

Media Query

```
<mj-style>
/* Media Query para dispositivos móviles */
@media only screen and (max-width: 480px) {
```

```

.header-info-text {
font-size: 12px !important;
text-align: center !important;
padding-left: 10px !important; /* Ajuste para móviles */
padding-right: 10px !important; /* Ajuste para móviles */
}
.header-view-link {
text-align: center !important;
padding-left: 10px !important; /* Ajuste para móviles */
padding-right: 10px !important; /* Ajuste para móviles */
}
h1 {
font-size: 24px !important;
padding: 10px 15px !important;
}
h2 {
font-size: 18px !important;
padding: 10px 15px !important;
}
/* Ajuste del tamaño de la fuente para todo el texto en mj-text */
mj-text {
font-size: 14px !important;
padding-left: 15px !important;
padding-right: 15px !important;
}
mj-button {
font-size: 16px !important;
padding: 10px 20px !important;
}
}
</mj-style>

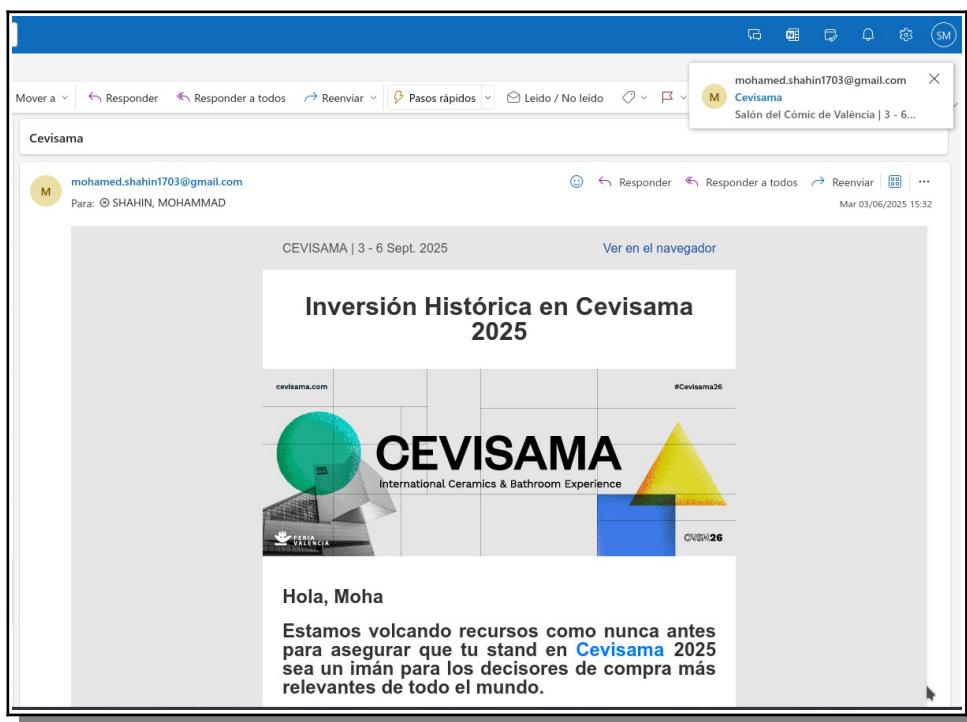
```

8. VERIFICAR COMPATIBILIDAD CON DIFERENTES CLIENTES DE CORREO

Hemos configurando un servicio de envío automatizado de correos que es compatible con diferentes correos.



EN OUTLOOK:



Ver en el navegador: <https://imagenes-publicas-mjml.s3.us-east-1.amazonaws.com/outputHTML/cevisama.html>