

Fiesta & Cultura

Aplicación web para la gestión de eventos culturales

Grupo 2:

Pablo Baudí Canales

Edilva García Álvarez

María González Pérez

Diego Piñol Puente

Elisabeth Segura Arcas

Usuarios de la aplicación

► Roles de usuario

- ✓ Gestor
- ✓ Moderador
- ✓ Consumidor registrado/no registrado

Requisitos funcionales

▶ Gestor

- ✓ Solicitar la modificación de datos, así como eliminación de cuenta.
- ✓ Gestionar ubicaciones.
- ✓ Gestionar y vender eventos.
- ✓ Comentar opiniones de usuarios.
- ✓ Imprimir listado de asistentes a evento.
- ✓ Comprar productos de publicación.
- ✓ Consultar estadísticas.

Requisitos funcionales

▶ **Moderador**

- ✓ Gestionar y confirmar usuarios, eventos y ubicaciones.
- ✓ Gestionar productos de publicación.
- ✓ Moderar contenido de los eventos y de la web.
- ✓ Consultar estadísticas.

Requisitos funcionales

▶ **Consumidor registrado**

- ✓ Ver contenidos de la web.
- ✓ Comentar eventos.
- ✓ Gestionar su perfil de usuario.
- ✓ Personalizar el aspecto de la interfaz.
- ✓ Comprar, personalizar e imprimir la entrada.
- ✓ Suscribirse al boletín de noticias.

Requisitos funcionales

▶ **Consumidor no registrado**

- ✓ Ver contenidos de la web.
- ✓ Registrarse como usuario ya sea consumidor o gestor.
- ✓ Suscribirse al boletín de noticias.

Modelo Navegacional

▶ **Arquitectura de servicios**

Se compone de 3 fachadas (dos remotas y una local)

- ✓ Aplicación de escritorio
 - Gestor y Moderador. Servicios remoto WCF
- ✓ Aplicación web
 - Consumidor. Servicio local

Aplicación de Escritorio

▶ Patrón MVP (Model View Presenter)

- ✓ Desacopla el modelo de la vista.

▶ Patrón de inyección de dependencia

- ✓ Utiliza el Presenter para implantar los datos de la interfaz en la vista.

▶ Patrón master template

- ✓ Ventana principal fija con un frame situado en la parte media inferior.

Aplicación Web

▶ Patrón MVC (Model View Controller)

- ✓ Proporciona una mayor abstracción de los datos permitiendo crear modelos basados en el servicio local y otros exclusivos para solucionar problemas en la web (ejemplo: carrito de compras).

▶ Patrón Master Template

- ✓ *Headers* y *footer* fijos en toda la aplicación.

▶ Patrón de inyección de dependencia

- ✓ Para hacer *scaffolding* con las vistas.

Pruebas

▶ Pruebas con Gherkin y Specflow

- ✓ Se emplea el BDD (Behaviour Driven Development) para realizar tests basados en los comportamientos reales del usuario y se utiliza para testear los métodos:
 - ✓ Customs
 - ✓ ReadFilter
 - ✓ ReadAll
 - ✓ CP (Componente de proceso).

Pruebas

▶ **Pruebas con MSTest**

- ✓ Las pruebas de código con MSTest fueron implementadas para las fachadas de los servicios WCF.
- ▶ Uso de software de integración continua Jenkins y Sonar para la cobertura de código.
- ▶ Un total de casi 300 pruebas.

Diseño de las vistas

▶ **Aplicación de escritorio**

✓ Uso de grids:

- 20 columnas y 10 filas (ventana central)
- 8 columnas y 9 filas (ventana interna de formularios)

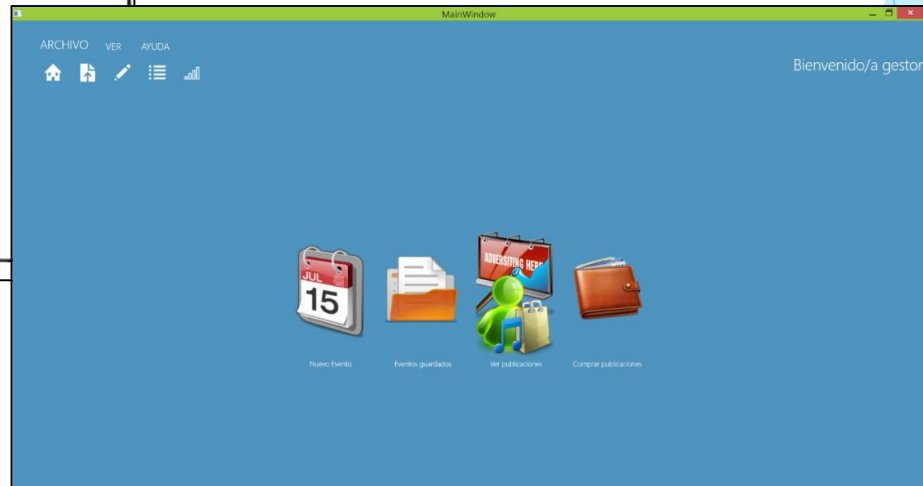
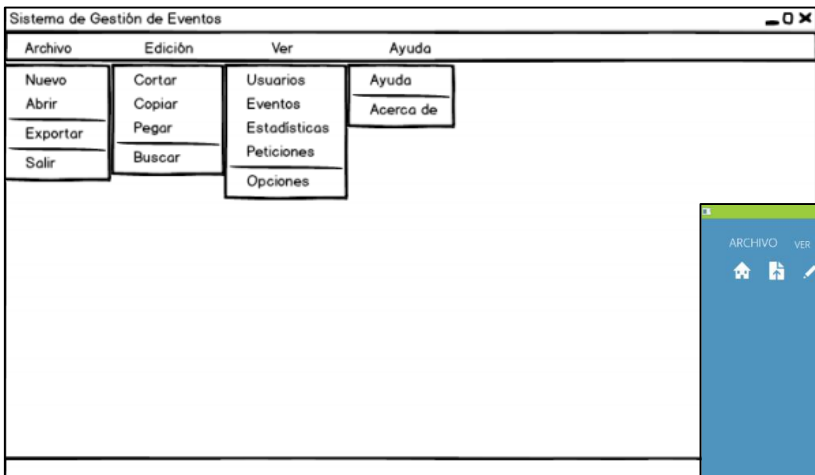
▶ **Aplicación web**

- ✓ Uso de un grid en la mayoría de ventanas de 1x10x1.
- ✓ Diseño web responsive.

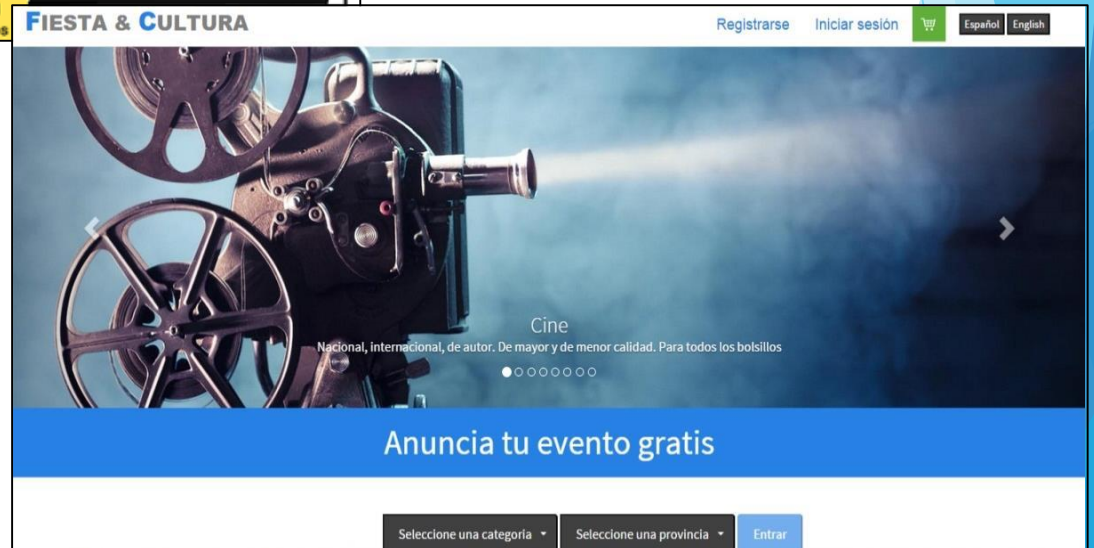
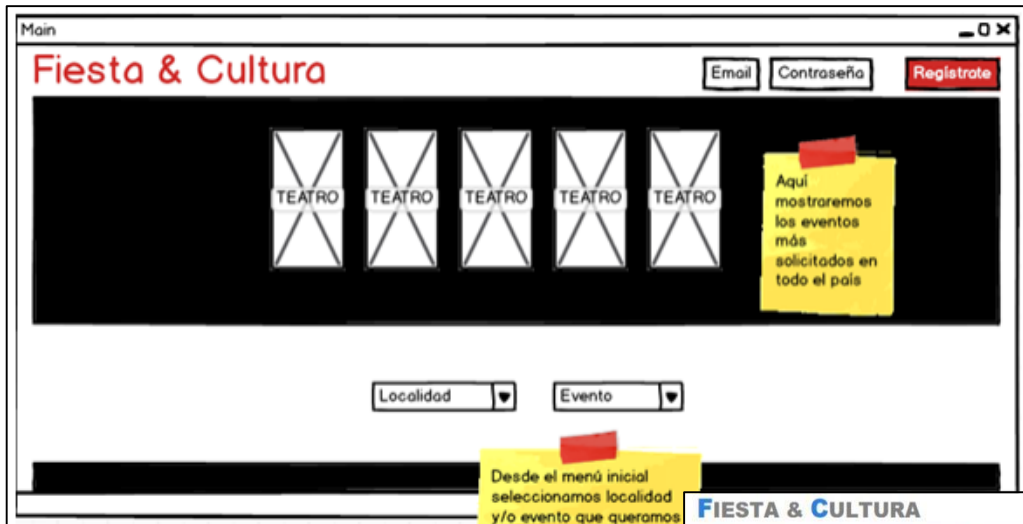
▶ **Pruebas con usuarios reales para mejorar el diseño.**

Diseño de las vistas

- ▶ **Uso de mockups para evaluar el diseño final de las aplicaciones**



Diseño de las vistas



Estadísticas

▶ **Gestor**

- ✓ Compras. Reflejan los productos de publicación en función de los años y los meses.
- ✓ Ventas. Muestran los beneficios y la asistencia a todos los eventos creados por el gestor.

▶ **Moderador**

- ✓ Concentra a los gestores de máximas y mínimas ventas.

▶ **¿Por qué se han elegido estas estadísticas?**

- ✓ Análisis de beneficios y ventas que permiten a los usuarios gestores mejorar su productividad y aumentar su proyección comercial.

Demos de las Aplicaciones

- ▶ A continuación, se presentan las demos de cada una de las aplicaciones para la gestión de eventos culturales en Fiesta & Cultura.
 - ▶ [Aplicación de Escritorio Gestor.](#)
 - ▶ [Aplicación de Escritorio Moderador.](#)
 - ▶ [Aplicación web.](#)