

ACOES



DOCUMENTO GENERAL DE REQUISITOS

SISTEMAS DE INFORMACIÓN PARA INTERNET



ACOES

DOCUMENTO GENERAL DE REQUISITOS

Contenido

INTRODUCCIÓN.....	3
REQUERIMIENTOS ESPECÍFICOS.....	3
Requisitos funcionales.....	3
1. GESTIÓN DE USUARIO:	3
2. GESTIÓN DE LA APLICACIÓN	4
Requisitos no funcionales.	7
EFICIENCIA.....	7
USABILIDAD	7
SEGURIDAD.....	7
SERVIDOR WEB.....	7
LENGUAJES UTILIZADOS	8

MIEMBROS DEL GRUPO

Andrés Valentín Suárez Mediavilla - 77425032V

Sergio González Sicilia - 77225653W

Ignacio Pascual Gutiérrez - 54236255R

Katia Moreno Berrocal - 79038916Z

Rocío Montalvo Lafuente - 03138701Y



INTRODUCCIÓN.

Este documento está elaborado con el propósito de presentar los requisitos que requiere la renovación del software perteneciente a ACOES, creando una aplicación para la gestión de niños/jóvenes desde que entran en el proyecto hasta que salen. Principalmente se quiere recoger y organizar la información relativa a la gestión de becas, CCJ, las casas Populorum y apadrinamiento.

El texto en rojo indica los requisitos que se han modificado con respecto a la anterior entrega: Tarea 2.

REQUERIMIENTOS ESPECÍFICOS.

Requisitos funcionales.

1. GESTIÓN DE USUARIO: la aplicación deberá tener 3 tipos de roles dependiendo del usuario correspondiente: usuario registrado, socio y administrador. Además de un usuario invitado del que no guardaremos información alguna.

1.1. Invitado:

- 1.1.1. Permiso de lectura.
- 1.1.2. Tiene opción a registrarse.
- 1.1.3. Tiene opción a iniciar sesión.

1.2. Role Socio:

- 1.2.1. Tiene permiso de lectura.
- 1.2.2. Tiene opción a cerrar sesión.
- 1.2.3. Un usuario podrá modificar sus propios datos.
(Se ha eliminado este requisito debido a que no pertenecía a los objetivos del proyecto)

1.3. Role Administrador:

- 1.3.1. Permiso de lectura.
- 1.3.2. Tiene opción a cerrar sesión.
- 1.3.3. Tiene opción a crear/modificar la información de los niños/jóvenes, becas, ccj, populorum y envíos.
- 1.3.4. Puede eliminar información de niños/jóvenes, becas, ccj, populorum y envíos.

**1.4. Role Usuario Registrado:**

- 1.4.1. Tiene permiso de lectura.
- 1.4.2. Tiene opción a cerrar sesión.
- 1.4.3. Tiene opción a volverse socio.

2. GESTIÓN DE LA APLICACIÓN**2.1. El sistema proporcionará un apartado para el registro.**

- 2.1.1. Debe contener un campo obligatorio Nombre de Usuario.
- 2.1.2. Debe contener un campo obligatorio Nombre (real).
- 2.1.3. Debe contener un campo obligatorio Primer Apellido.
- 2.1.4. Debe contener un campo opcional Segundo Apellido.
- 2.1.5. Debe contener un campo obligatorio E-mail.
- 2.1.6. Debe contener un campo obligatorio contraseña.
- 2.1.7. Debe contener un campo obligatorio Fecha de Nacimiento.
- 2.1.8. Debe contener un campo obligatorio para aceptar las políticas de privacidad de datos.
- 2.1.9. Debe contener un campo obligatorio para aceptar las políticas de protección de datos.
(Se han eliminado estos dos requisitos debido a que no pertenecían a los objetivos del proyecto)
- 2.1.10. Debe contener un campo obligatorio DNI.
- 2.1.11. Debe contener un campo obligatorio Teléfono.

2.2. Deben existir distintos apartados: Gestión de becas, CCJ, Populorum y apadrinamiento de niños.**2.2.1. Gestión de becas.**

- 2.2.1.1. Los administradores podrán crear nuevas becas.
- 2.2.1.2. Los administradores podrán modificar becas ya existentes.
- 2.2.1.3. Los administradores podrán eliminar becas.
- 2.2.1.4. Los administradores podrán visualizar las becas ya existentes
- 2.2.1.5. Los socios podrán visualizar las becas ya existentes relacionadas con sus niños apadrinados.
- 2.2.1.6. El sistema listará todas las becas ya existentes.
- 2.2.1.7. El sistema ordenará las becas por orden de antigüedad.
(Se ha eliminado este requisito debido a que no lo creíamos conveniente para el proyecto)



2.2.1.8. El sistema dispondrá a los socios la función para donar dinero para becas.

(Se ha eliminado este requisito debido a que no pertenecía a los objetivos del proyecto).

2.2.2. CCJ.

2.2.2.1. El sistema listará todos los Centros de Capacitación juvenil existentes e información relativa a cada uno.

2.2.2.2. Los administradores podrán añadir nuevos centros.

2.2.2.3. Los administradores podrán eliminar CCJ.

2.2.2.4. Los administradores podrán visualizar los CCJ ya existentes.

2.2.2.5. Los socios podrán visualizar los CCJ de sus apadrinados.

2.2.2.6. El sistema listará todas los CCJ ya existentes.

2.2.2.7. El sistema ordenará los CCJ por orden de antigüedad.
(Se ha eliminado este requisito debido a que no lo creíamos conveniente para el proyecto)

2.2.2.8. Los administradores añadirán a los niños a su CCJ correspondiente.

2.2.3. Populorum.

2.2.3.1. El sistema listará todas las casas Populorum existentes e información relativa a cada una.

2.2.3.2. Los administradores podrán añadir nuevas casas Populorum.

2.2.3.3. Los administradores podrán eliminar casas Populorum.

2.2.3.4. Los administradores podrán visualizar las casas Populorum ya existentes.

2.2.3.5. Los socios podrán visualizar las casas Populorum de sus apadrinados.

2.2.3.6. El sistema listará todas las casas Populorum ya existentes.

2.2.3.7. El sistema ordenará las casas populorum por orden de antigüedad.

(Se ha eliminado este requisito debido a que no lo creíamos conveniente para el proyecto)

2.2.3.8. El administrador podrá añadir niños a su respectiva casa Populorum.



2.2.4. Apadrinamiento de niños (Gestión de niños).

2.2.4.1. El sistema permitirá que un socio pueda apadrinar a uno o varios niños/jóvenes.

2.2.4.2. El sistema ordenará los niños por orden de antigüedad descendiente respecto al último apadrinamiento.

(Se ha eliminado este requisito debido a que no lo creíamos conveniente para el proyecto)

2.2.4.3. El sistema listará todos los niños posibles para apadrinar. (Se ha cambiado este requisito por un sistema automático de apadrinamiento tal y como se nos pidió en la corrección de la tarea 2)

2.2.4.4. Los administradores podrán añadir nuevos niños para ser apadrinados.

2.2.4.5. Los socios podrán visualizar a los niños que hayan apadrinado.

2.3. El sistema debe tener un inicio. Será lo primero que se le muestre al usuario.

2.3.1. Aparecerá la información de la organización resumida.

2.3.2. Deberá tener un menú o un desplegable para acceder a los apartados que describimos en el punto 2.2.

2.5. Envíos.

2.5.1 Los socios podrán enviar objetos a su/s niño/s apadrinado/s.

2.5.2 Los administradores podrán crear y leer **envíos**.

2.5.2 El sistema mostrará en todo momento el estatus de dichos envíos, siendo: recibido, entregado (al niño/joven), pendiente.



Requisitos no funcionales.

EFICIENCIA.

- 1.1. La aplicación deberá realizar cualquier transacción antes de 5 segundos.
- 1.2. La página deberá cargar antes de 2 segundos.
- 1.3. Deberá estar optimizada mediante herramientas SEO.
(Hemos eliminado dicho requisito).

USABILIDAD

- 2.1. La web deberá ser intuitiva y sencilla.
- 2.2. La aplicación deberá tener un diseño responsive.
- 2.3. El usuario no deberá dedicar más de 5 minutos para saber desenvolverse con soltura en la página.
- 2.4. La página web deberá implementar un apartado de contacto con los administradores.
- 2.5. La web deberá ser compatible con todos los navegadores web.

SEGURIDAD

- 3.1. Las contraseñas deberán estar encriptadas.
- 3.2. El sistema debe tener un registro de los errores ocasionados.
(Hemos eliminado dicho requisito, los errores están controlados pero no se lleva un registro de ellos).
- 3.3. La página web usará el protocolo HTTPS en vez de HTTP.
- 3.4. La contraseña de los usuarios deberá tener una longitud mínima de 8 caracteres, además de contener letras y números, como así lo dicta la política de contraseñas en seguridad avanzada.
- 3.5. El sistema debe mostrar una información distinta para cada tipo de role.
Explicación: Esto es totalmente necesario porque no cualquiera puede ver la información de los niños que pueden ser apadrinados.

SERVIDOR WEB

- 4.1. La base de datos estará en un gestor de contenido online (ej: aws).
(La BD se encuentra dentro del equipo local finalmente)



LENGUAJES UTILIZADOS

- 5.1. HTML5
- 5.2. CSS3 (Bootstrap)
- 5.3. Java
- 5.4. SQL