



**sistema
control
garantías**

Santiago Cámara Sánchez

Desarrollo de aplicaciones multiplataforma

IES Abastos. Curso 2020/21. Grupo 7U. 9 de junio de 2021

Tutor individual: Eduardo González Sanz



SE HA AVERIADO LA LAVADORA

¿Cuándo se compró y cuanto costó?

¿Estará todavía en garantía?

¿Dónde está la factura de compra?





HAY QUE CONTRATAR UN SEGURO DE HOGAR

¿Por cuánto valoro el contenido?

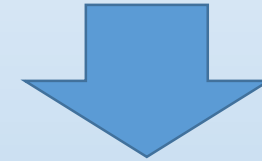




Presentación



DOS PROBLEMAS



UNA SOLUCIÓN

SCG – Sistema de control de garantías



FUNCIONES PRINCIPALES DE SCG

1. Gestión de artículos/productos
2. Control del estado de las garantías
3. Inventario por categorías



FUNCIONES ADICIONALES DE SCG

1. Gestión de planes de mantenimiento periódico
2. Almacenamiento de facturas y tickets
3. Archivo de Manuales de usuario

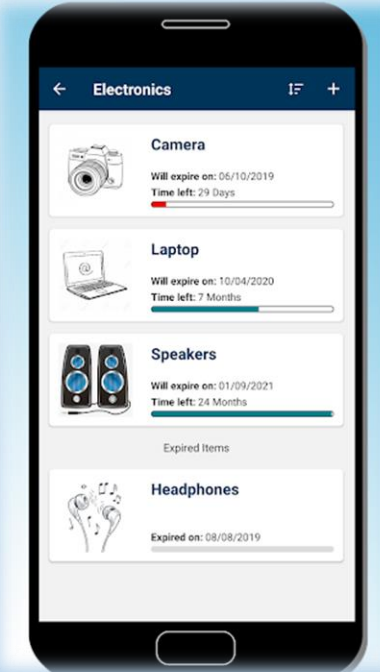
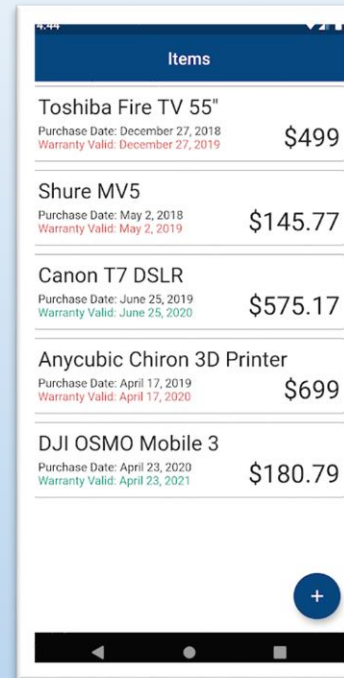
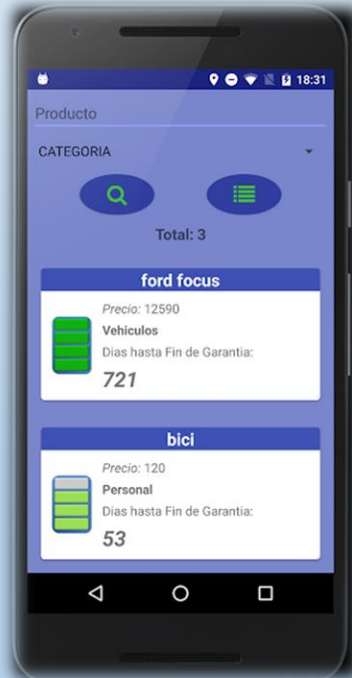


Análisis de la competencia

- Inexistencia de aplicaciones para escritorio de esta temática
- Algunas opciones en Android, pero solo en inglés o con escasa funcionalidad



Análisis de la competencia





Diseño de la aplicación

HERRAMIENTAS Y SOLUCIONES TECNOLÓGICAS

- Node.js (JavaScript) + Framework Electrón
- Plantilla de diseño Photon kit
- IDE Microsoft visual Code

DESARROLLO MULTIPLATAFORMA

- Windows
- Linux
- Mac OS



Diseño de la aplicación

PERSISTENCIA DE DATOS

```
{
  "0": {
    "nombre": "Sin categoría",
    "icono": "sin_categoria.png"
  },
  "1": {
    "nombre": "Audio",
    "icono": "icono_cat_1.png"
  },
  "2": {
    "nombre": "Cocina",
    "icono": "icono_cat_2.png"
  } ...
}
```

- Ficheros JSON
Objetos identificados por clave



Diseño de la aplicación

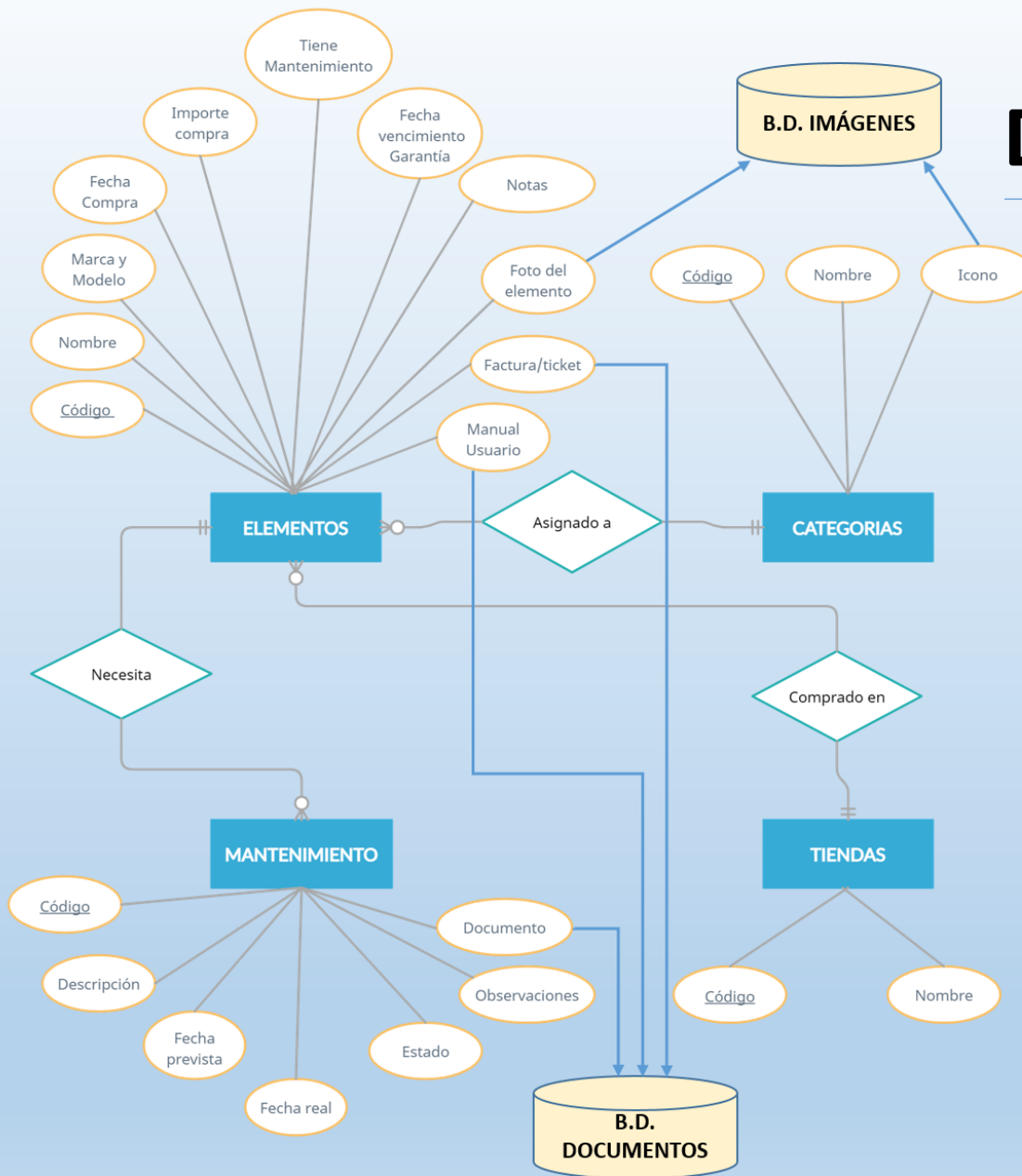
PLANIFICACIÓN

Fase 1	Fase 2	Fase 3
<ul style="list-style-type: none">• Gestión elementos• Control del estado de garantías• Inventario valorado	<ul style="list-style-type: none">• Gestión de categorías• Gestión de tiendas• Tickets y facturas• Manuales de usuario	<ul style="list-style-type: none">• Planes de mantenimiento periódico• Gestión de copias de seguridad



Desarrollo del proyecto

MODELO ENTIDAD RELACIÓN





REGLAS DE NEGOCIO ELEMENTOS (1)

- Nombres de elementos únicos y no vacíos o en blanco
- Atributos obligatorios: nombre, categoría, tienda, fecha de compra.
- Categoría por defecto "Sin categoría".
- Tienda por defecto "otras tiendas".
- La fecha de vencimiento de la garantía será calculada en base a la fecha de compra.



REGLAS DE NEGOCIO ELEMENTOS (2)

- Si no tiene garantía, se considerará la fecha de compra como fecha de vencimiento de garantía.
- Almacenamiento opcional de imagen (jpg, jpeg o png)
Por defecto "no disponible"
- Almacenamiento opcional de factura/ticket y manual de usuario (pdf, jpg, jpeg o png)



REGLAS DE NEGOCIO CATEGORÍAS Y TIENDAS

- Nombres de únicos y no vacíos o en blanco
- Se garantiza la integridad referencial
- Se puede asignar un icono a las Categorías (ico o png)
- La categoría por defecto "Sin categoría" no puede eliminarse ni modificar su icono.
- La Tienda por defecto "otras tiendas" no puede eliminarse.
- Existirán datos predefinidos de categorías y tiendas.



REGLAS DE NEGOCIO PLANES DE MANTENIMIENTO

- Son opcionales. Si lo hay pueden ir de 1 a 50 operaciones de mantenimiento por elemento.
- Las fechas se calculan tomando como base a la fecha de compra y añadiendo un intervalo periódico.
- Se consultan ordenados por fecha prevista. Filtros:
 - todas las fechas / desde los últimos 90 días en adelante / futuros
 - pendientes de realizar / ya realizados.
 - por nombre del elemento



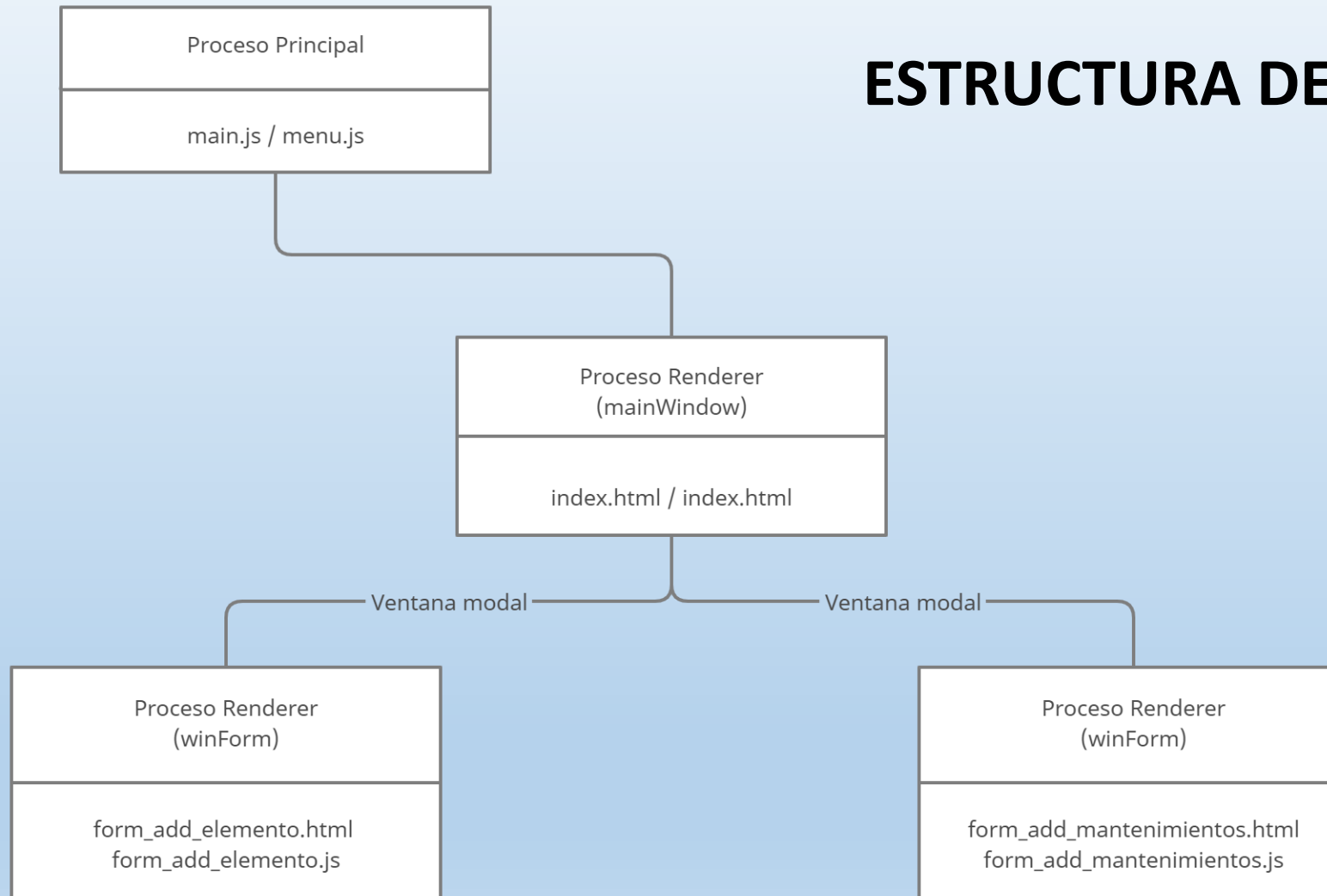
REGLAS DE NEGOCIO GENERALES

- El inventario podrá consultarse en su totalidad o filtrado por categoría.
- La lista de elementos se podrá consultar por orden alfabético (por defecto), por fecha de vencimiento de garantía o por orden de alta en la aplicación.
- La aplicación gestionará las copias de seguridad que estarán identificadas por su fecha y hora y descripción proporcionada por el usuario.



Desarrollo del proyecto

ESTRUCTURA DE PROCESOS





Desarrollo del proyecto

ESTRUCTURA DE PROCESOS

Nombre	Descripción
main.js	Proceso principal
menu.js	Gestión del menú de la aplicación
renderer.js	Gestión de la ventana principal
form_add_itemregister.js	Gestión ventana modal “añadir elementos”
form_gest_mantenimientos.js	Gestión ventana modal “mantenimientos”
Módulo GUI.js	Gestión interfaz de usuario ventana principal
Módulo itemStorage.js	Definición de objetos de elementos y sus métodos
Módulo mant.js	Funciones comunes de mantenimiento periódico
Módulo util.js	Utilidades comunes a todos los procesos



Desarrollo del proyecto

DISEÑO GRÁFICO

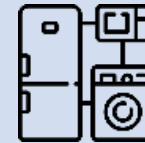
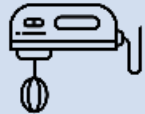
ICONO APLICACIÓN Y BARRA DE HERRAMIENTAS	LOGOTIPO DE LA APLICACIÓN
	 sistema control garantías



Desarrollo del proyecto

DISEÑO GRÁFICO

ICONOS DE CATEGORÍAS





Desarrollo del proyecto

DISEÑO GRÁFICO - COHERENCIA USO COLORES

- Semáforo para fechas de vencimiento de garantía y fechas previstas de mantenimiento

- **FECHAS VENCIDAS**
- **FECHAS QUE VENCEN EN 90 DÍAS O MENOS**
- **FECHAS VENCERÁN EN MÁS DE 90 DÍAS**



Desarrollo del proyecto

DISEÑO GRÁFICO - COHERENCIA USO COLORES


- Botones con idéntico color para funciones similares

Blanco y negro	Naranja	Rojo	Azul	Verde
Navegación	Cancelar o salir	Eliminar	Acciones positivas	Ver documentos
 Copias de seguridad	 Cancelar	 Eliminar	 Añadir elemento	 Visualizar manual



DISEÑO GRÁFICO – BOTONES AUTO-EXPLICATIVOS

- Todos los botones tienen un icono según su función.
- La mayoría tienen texto o, en caso contrario, una ayuda emergente al poner el ratón encima.

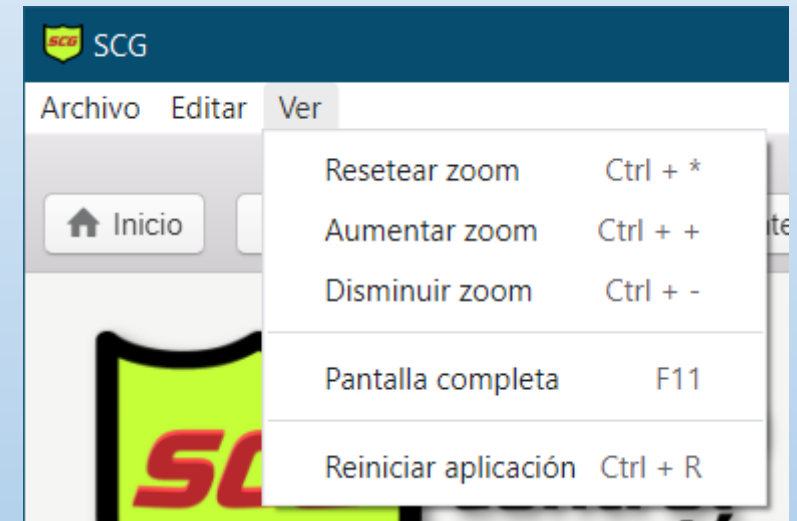
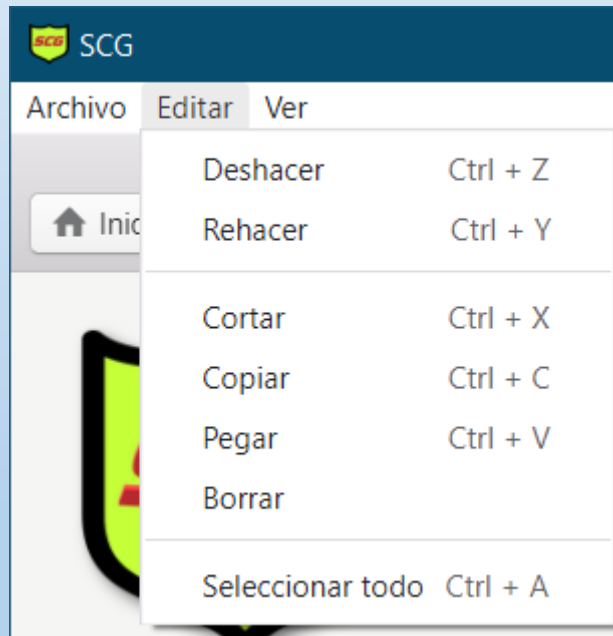
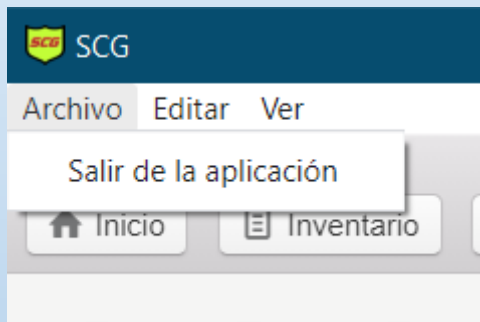
Botón auto-explicativo	Botones sin texto	Botón sin texto al poner el ratón encima
		



Desarrollo del proyecto

INTERFAZ DE USUARIO - MENÚS

- Se han creado menús para acciones propias de una aplicación de escritorio.

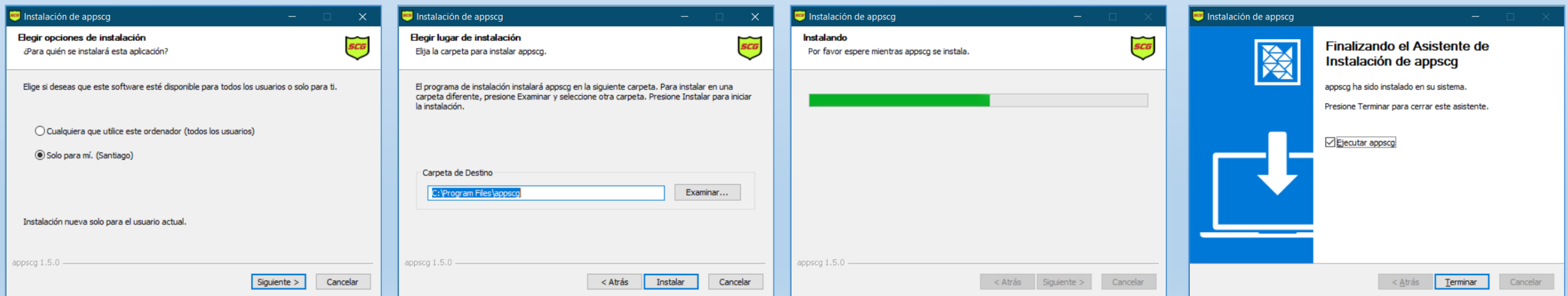




Desarrollo del proyecto

IMPLANTACIÓN Y DISTRIBUCIÓN

- Se ha implementado un paquete de instalación para Windows.
- También una versión portable que no requiere instalación.





IMPLANTACIÓN Y DISTRIBUCIÓN

- No se ha podido probar en Mac OS por carecer del equipo adecuado.
- En Linux ha sido imposible crear un paquete de instalación por problemas técnicos.
- No obstante, si se ha podido comprobar que la aplicación se ejecuta en Linux (en máquina virtual) mediante comandos.



Desarrollo del proyecto

LICENCIA

- Creative Commons Atribución/Reconocimiento - No Comercial - Compartir Igual 4.0 Internacional (CC BY-NC-SA)

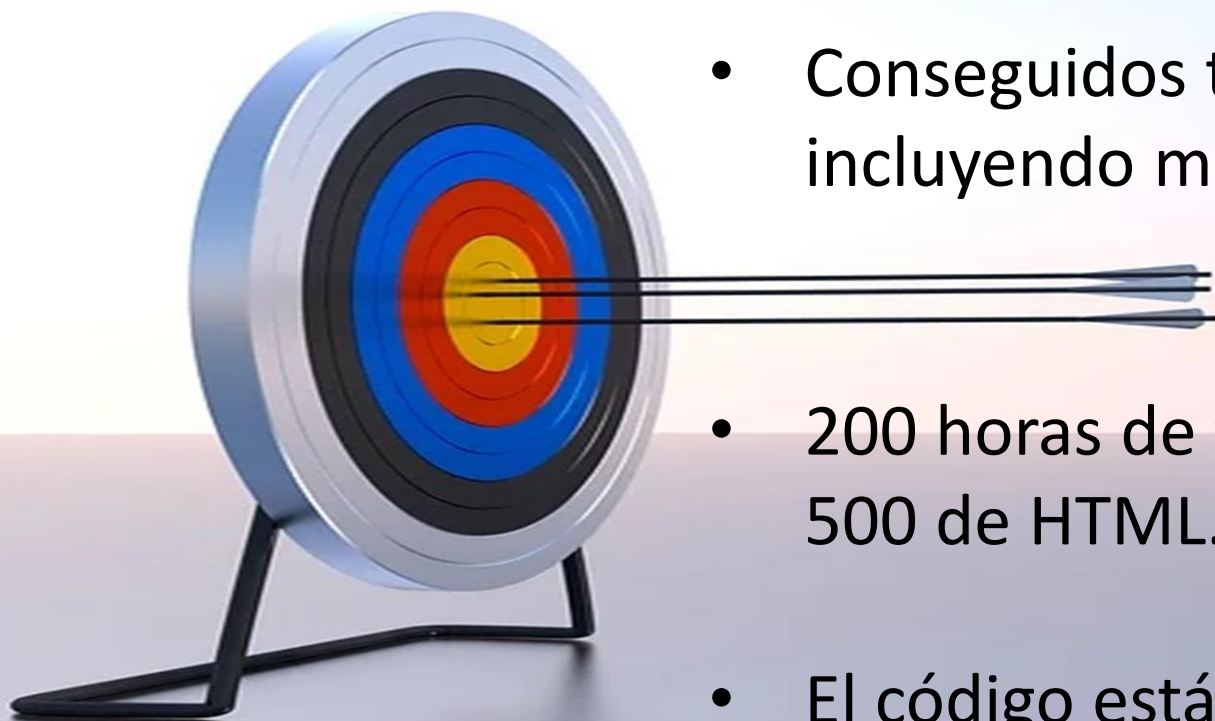


- Se debe atribuir o reconocer al creador.
- Solo se permiten usos no comerciales de la aplicación.
- Cualquier adaptación del software debe compartirse en los mismos términos.



Evaluación y conclusiones finales

EVALUACIÓN EJECUCIÓN DEL PROYECTO



- Conseguídos todos los objetivos iniciales, incluyendo mejoras sugeridas por usuarios
- 200 horas de trabajo, 3.000 líneas de JavaScript y 500 de HTML.
- El código está redactado de forma modular para facilitar al máximo su mantenimiento.



Evaluación y conclusiones finales

PROPUESTA DE MEJORAS

- Planes de mantenimiento: mejorar la personalización de las fechas previstas de las operaciones.
- Almacenar información en la nube.
- Desarrollar la versión para dispositivos móviles.
- Definición de niveles de usuario.
- Crear un paquete de instalación diferenciado para Linux y Mac OS.



DEMOSTRACIÓN DE LA APLICACIÓN



**sistema
control
garantías**