

WEB DE PROTOCOLOS HSJD



Protocolos

HOSPITAL SAN JUAN DE DIOS
DEL ALJARAFE

Índice

- 1. Introducción**
- 2. Requisitos hardware y software**
- 3. Guía de instalación y puesta en marcha**
- 4. Funcionamiento general (pantallas y flujos)**
- 5. Procesos lógicos o complejos
(fórmulas, código Power Apps, integración SharePoint)**
- 6. Interfaz de usuario y control de errores**
- 7. Patrón de arquitectura (esquemático)**
- 8. Dificultades encontradas**
- 9. Conclusiones y mejoras futuras**

1. Introducción

1.1. Descripción:

Este Trabajo de Fin de Grado aborda la necesidad de mejorar el acceso a la documentación interna de tipo protocolo en un entorno hospitalario. Actualmente, los empleados deben navegar por estructuras de carpetas complejas o depender de sistemas poco intuitivos para encontrar los documentos que les corresponden, lo que puede generar retrasos o errores.

La solución propuesta es una aplicación desarrollada con Microsoft Power Apps que se conecta a listas de SharePoint. Su objetivo principal es permitir a cada empleado acceder únicamente a los protocolos que le corresponden, según su categoría profesional y servicio. Para ello, la app incorpora un sistema de filtrado personalizado y una visualización directa de los documentos en formato PDF.

El público destinatario de la app son los empleados del **Hospital San Juan de Dios del Aljarafe** que cuentan con un perfil en SharePoint, permitiéndoles consultar sus protocolos de forma rápida, segura y desde cualquier dispositivo con conexión a internet.

Con esta solución se busca optimizar el tiempo de búsqueda de documentos, mejorar la experiencia de usuario y asegurar el acceso controlado a la información según el perfil del trabajador.

1.2. Distribución del contenido:

Tras la introducción, que plantea el problema y los objetivos de la app, se describen los requisitos y se guía al lector en la instalación. A continuación se detalla el diseño de pantallas y flujos de datos, seguido de un análisis de la lógica interna (fórmulas y código). Después se explica la interfaz y la gestión de errores, se presenta el esquema arquitectónico, se exponen los retos superados y, finalmente, se cierran con las conclusiones y propuestas de mejora futura.

2. Requisitos hardware y software

2.1. Requisitos de hardware:

- **Para usuarios finales:** Al ser una solución completamente alojada en la nube (Microsoft 365), los usuarios solo necesitan dispositivos estándar con conexión a Internet. Se ha verificado el funcionamiento en:
 - Ordenadores de sobremesa (PCs) y portátiles típicos del entorno hospitalario.
 - Dispositivos móviles (smartphones y tablets).
Un equipo moderno con 8GB de RAM y 4GB de espacio libre en disco es suficiente para garantizar un acceso fluido a través del navegador web o la aplicación móvil.
- **Para desarrollo:** El desarrollo se realizó con éxito en un equipo con las siguientes características (demostrando que no se requieren recursos excesivos):
 - Procesador: Intel Core i3-10100 @ 3.60 GHz.
 - Memoria RAM: 8.00 GB.
 - Almacenamiento: 238 GB.
 - Tarjeta gráfica: Gráficos integrados Intel UHD (128 MB dedicados). Este equipo cumple holgadamente los requisitos para utilizar Power Apps Studio a través del navegador.

2.2 Requisitos de software:

- **Sistemas Operativos compatibles (verificados y recomendados):**
 - Windows 11.
 - Android 11 o superior.

(Nota: Power Apps también es compatible con otras versiones de Windows, macOS e iOS, aunque la verificación específica se realizó en los SO mencionados).

- **Acceso a la aplicación:**
 - **Navegadores Web:** La aplicación es accesible desde la mayoría de navegadores web modernos. Se ha probado específicamente con Opera y Google Chrome. Se recomienda utilizar las versiones más recientes de estos o de otros navegadores

compatibles con Power Apps (como Microsoft Edge o Mozilla Firefox).

- **Aplicación Móvil de Power Apps:** Para una experiencia óptima en dispositivos móviles (smartphones y tablets), se recomienda utilizar la aplicación oficial Power Apps (disponible en Google Play Store y Apple App Store). No obstante, el acceso también es posible directamente a través del navegador web del dispositivo móvil.
- **Software Adicional:** Es imprescindible que el navegador web utilizado incorpore o tenga acceso a un visor de PDF estándar, ya que la aplicación presenta los protocolos hospitalarios en este formato.

2.3. Plataforma en la nube y licencias:

- **Plataforma:** La aplicación reside completamente en Microsoft 365, utilizando los servicios SharePoint Online (para el almacenamiento y gestión estructurada de los protocolos y datos) y Power Apps (para la interfaz de usuario y la lógica de la aplicación).
- **Licencias de usuario:** Para acceder y utilizar la aplicación, cualquier empleado del hospital necesita únicamente disponer de una cuenta corporativa válida de Microsoft 365 asignada por el centro. Esta cuenta debe tener la licencia mínima de Microsoft 365 que incluya los derechos de acceso a SharePoint Online y al uso de aplicaciones creadas con Power Apps (comúnmente incluidas en planes como Microsoft 365 F3, E3 o superiores). Los mismos permisos de cuenta permiten editar o crear contenido en las listas de SharePoint subyacentes si el usuario tiene los roles adecuados asignados.
- **Licencias y herramientas de desarrollo:** El desarrollo requirió una licencia de Microsoft 365 que proporcionará acceso a Power Apps con capacidad de creación (como las incluidas en planes E3, E5 o la licencia específica de desarrollador). Las herramientas principales utilizadas fueron:
 - **Power Apps Studio (accedido vía web):** Para el diseño y desarrollo de la interfaz y lógica de la aplicación.
 - **SharePoint Online (interfaz web):** Para la creación y configuración de las listas y bibliotecas que almacenan la información.

- **Power Automate (web - para futuras integraciones):** Aunque no implementadas en la versión actual, se exploró para automatizar flujos de trabajo asociados.

2.4. Conectividad y otros requisitos:

- **Red:** Es necesaria una conexión a Internet estable para acceder a la aplicación y a los protocolos almacenados en SharePoint Online. Si bien no se requiere un ancho de banda extremadamente alto, una conexión razonable garantizará una carga rápida de los documentos PDF.
- **Actualizaciones:** Se recomienda mantener los navegadores web y sistemas operativos actualizados (dentro de las versiones compatibles mencionadas) para asegurar la mejor compatibilidad y seguridad.
- **Acceso y Seguridad:** La aplicación es accesible directamente desde Internet a través del portal de Power Apps o SharePoint Online, sin necesidad de VPN u otros accesos especiales a la red interna del hospital. El acceso está protegido por la autenticación estándar de Microsoft 365 (Azure AD) gestionada por el centro. No se requieren configuraciones especiales de firewall más allá de permitir el acceso a los servicios de Microsoft 365.
- **Dispositivos:** Como se mencionó, la aplicación está diseñada para ser utilizada principalmente en los ordenadores de sobremesa y portátiles asignados a los diferentes puestos y servicios del hospital, así como en los dispositivos móviles corporativos proporcionados por el centro.





2.5. Resumen:

En definitiva, los requisitos hardware y software de la aplicación de protocolos son mínimos y están ampliamente cubiertos por la infraestructura tecnológica típica de un entorno hospitalario moderno. La solución aprovecha las inversiones existentes en Microsoft 365, requiriendo únicamente dispositivos estándar con conexión a Internet y las licencias corporativas de Microsoft 365 ya asignadas al personal. Esto elimina la necesidad de hardware especializado, servidores adicionales o licencias complejas, facilitando su implantación y uso generalizado.

3. Guía de instalación y puesta en marcha

Dado que la solución es nativa de la nube, el proceso requiere sólo 5 minutos y es responsabilidad del personal de informática del centro.

3.1. Requisitos previos

Elemento	Detalle
 Licencias	Cuenta de administrador en Microsoft 365 con licencia que incluya: SharePoint Online y Power Apps Premium
 Permisos	Rol de "Propietario" en: <ul style="list-style-type: none">- El sitio SharePoint donde residen las listas- Entorno Power Apps donde se aloja la aplicación
 Infraestructura	Listas de SharePoint preconfiguradas con los campos necesarios (ej: "Protocolos", "Categorías")
 Cliente	Navegador web actualizado (Chrome, Edge)

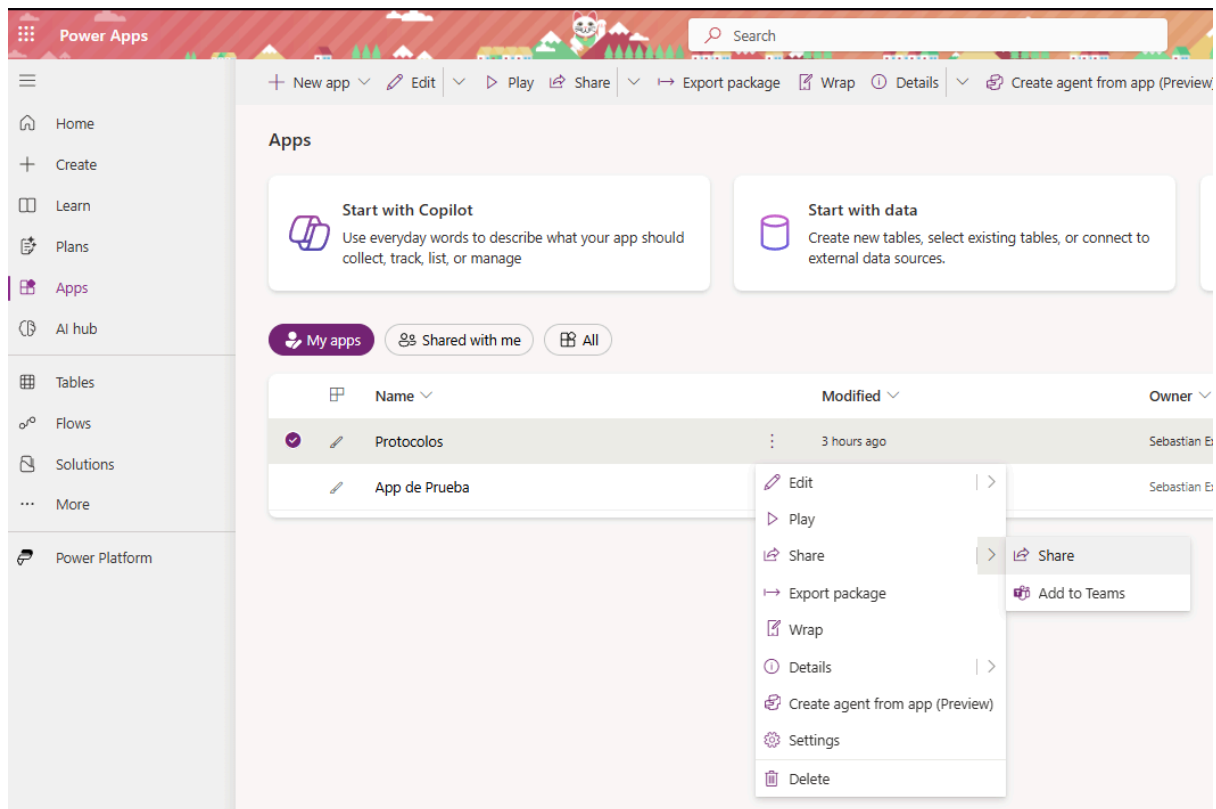
3.2. Procedimiento de instalación (Paso único)

1. Acceder a Power Apps Studio

- Inicie sesión en make.powerapps.com con cuenta de administrador.
- Seleccione el entorno correcto ("Protocolos").

2. Compartir la aplicación

- En la sección "Apps", localice la aplicación "Protocolos".
- Haga clic en los \cdots (menú contextual).
- Seleccione "Share" > "Share" nuevamente.



3. Asignar usuarios

- Ingrese los grupos de seguridad o correos de los empleados (ej: todos@hospital.es).
- Asegúrese que el permiso sea "Usuario" (acceso sin capacidad de editar el diseño).
- Haga clic en "Aceptar".

Share "Protocolos" ... ✕

i This app contains data that may require additional permissions for users to be able to play the app.
[Review data access requirements](#)

Add a name, group, or email

Include a message or image for an email notification to be sent.

0 / 500

Manage access Copy link Share

3.3. Configuración post-instalación

- Permisos en SharePoint:

Los usuarios heredarán acceso a las listas según su rol en el sitio. No se requiere configuración adicional.

- Acceso para editores:

Personal administrativo que deba modificar protocolos necesita:

- Rol **Colaborador** en las listas de SharePoint asociadas.
- Permiso **Editor** en la app.


3.4. Verificación del funcionamiento

Realice estas pruebas con un usuario estándar:

1. Abra powerapps.microsoft.com.
2. Inicie sesión con cuenta de empleado (ej: enfermeria@hospital.es).
3. Ejecute la aplicación "Protocolos" y compruebe:
 - ☒ Carga correcta de la interfaz.
 - ☒ Visualización de documentos PDF.
 - ☒ Funcionamiento de filtros/búsqueda.



Iniciar Sesión



Solución de errores frecuentes:

- "No se cargan los protocolos" → Verificar permisos del usuario en las listas SharePoint.
- "Acceso denegado a la app" → Revisar si la licencia M365 incluye Power Apps.

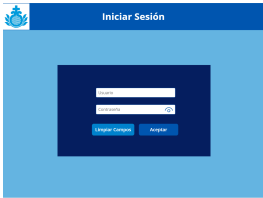

4. Funcionamiento general (pantallas y flujos)



4.1. Contexto y objetivo

La aplicación centraliza los protocolos clínicos del hospital en formato PDF, permitiendo:

- Acceso rápido filtrado por puesto/categoría del empleado.
- Seguimiento de lecturas (protocolos no leídos son prioritarios).
- Monitoreo de cumplimiento (solo para coordinadores de enfermería).

4.2. Pantallas principales

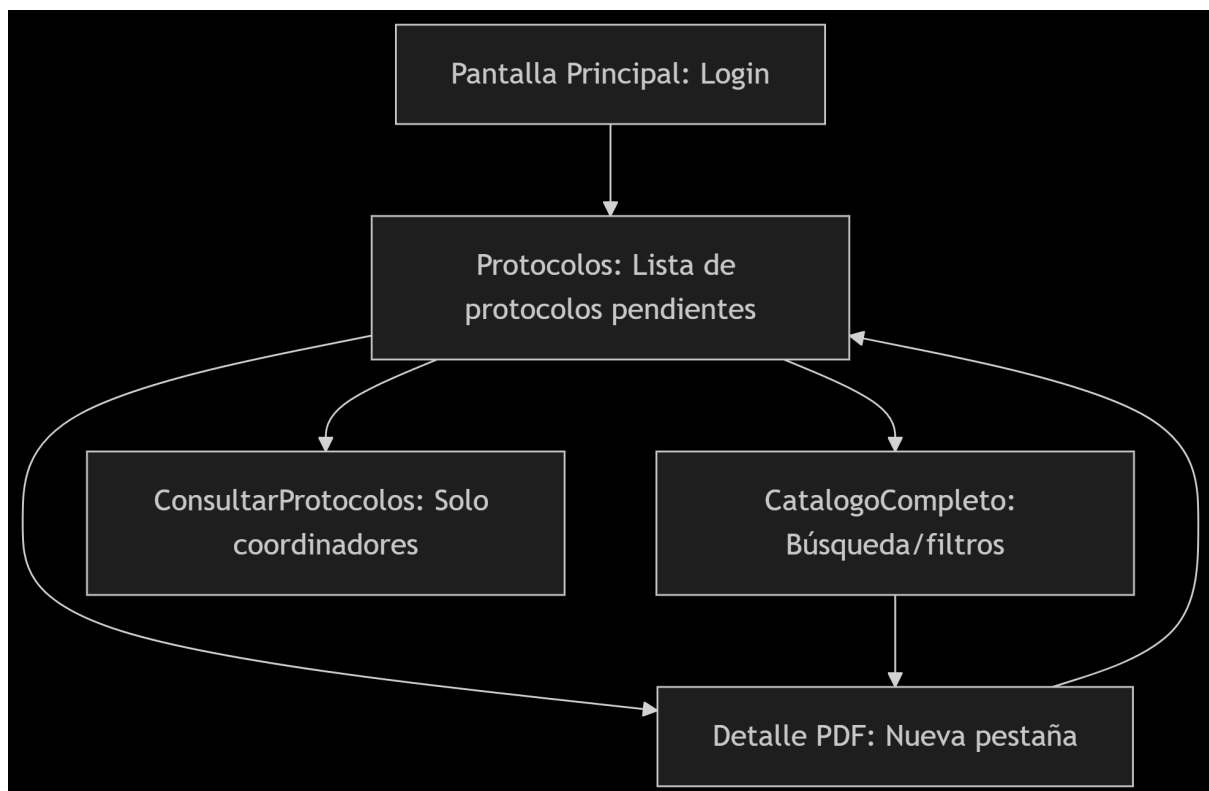
Pantalla	Elementos clave	Función principal	Interfaz
Principal	<div>- Logo del hospital</div> <div>- Campos de texto (usuario/contraseña)</div> <div>- Botón "Iniciar sesión"</div>	Autenticación inicial	
Protocolos	<div>- Galería dinámica de protocolos</div> <div>- Desplegables (puesto, prioridad)</div> <div>- Botones "Abrir" (marcan como leído)</div> <div>- Iconos de navegación</div>	Visualización y gestión de protocolos	

Catalogo Completo	<ul style="list-style-type: none"> - Barra de búsqueda superior - Galerías con filtros combinables - Botones de categorías 	Búsqueda avanzada	
Consultar Protocolos	<ul style="list-style-type: none"> - Campo de texto (ingresar usuario) - Gráficos de progreso - Botón "Generar reporte" 	Auditoría de lecturas (solo coordinadores)	

Nota técnica:

- Las galerías muestran protocolos como tarjetas interactivas (diseño visual sobre datos de SharePoint).
- Los botones "Abrir" actualizan el estado "leído" en SharePoint automáticamente.

4.3. Flujo de navegación principal



4.4. Casos de uso paso a paso

Caso 1: Acceso y visualización de protocolos

1. Iniciar sesión:

- El empleado ingresa credenciales en la pantalla Principal.
- Al validarse, la app redirige automáticamente a Protocolos.

2. Filtrar protocolos pendientes:

- La galería muestra solo protocolos no leídos correspondientes a su puesto.
- Opcional: Usar desplegables para filtrar por prioridad o categoría.

3. Abrir un protocolo:

- Clic en "Abrir" → El PDF se carga en una nueva pestaña del navegador (visor nativo con zoom/búsqueda).
- El sistema marca automáticamente el protocolo como "leído".

Caso 2: Búsqueda avanzada

1. Navegar a CatalogoCompleto desde el menú.

2. Combinar filtros:

- Seleccionar servicio (ej: Urgencias) + prioridad (ej: Alta).
- Escribir palabras clave en la barra de búsqueda (ej: "RCP").

3. Clic en un protocolo → Visualización PDF (misma lógica que Caso 1).

Caso 3: Auditoría de lecturas

(Exclusivo para coordinadores de enfermería)

1. Acceder a ConsultarProtocolos desde el menú.

2. Ingresar el usuario del empleado auditar.

3. Clic en "Buscar" → Muestra gráficos de % de protocolos leídos.

4.5. Contexto técnico simplificado

- Filtrado inteligente:

La app carga desde SharePoint solo los protocolos asignados al puesto + categoría del usuario.

- Ejemplo: Un enfermero de Urgencias verá sólo protocolos de su servicio no leídos.

- Interacción con SharePoint:

- Al pulsar "Abrir", la app:
 1. Actualiza la lista "T_Protocolos_vistos" en la lista de SharePoint.
 2. Recupera la URL del PDF desde la biblioteca de documentos.

- **Seguridad:**

El acceso a ConsultarProtocolos se bloquea si el usuario no tiene categoría "Coordinador de Enfermería".

5. Procesos lógicos o complejos (fórmulas, código Power Apps, integración SharePoint)

5.1. Sincronización de usuarios via VBScript

Para obtener los datos actualizados de la lista T_Usuarios_LDAP primero ejecuto un script el cual funciona de la siguiente manera:

1. Lee el contenido de un Excel que se me proporciona todos los días el cual contiene todos y cada uno de los contratos activos en el hospital.
2. Elimina los duplicados, los ordena y comprueba los datos con los existentes en mi base de datos en Access llamada ProtocolosFinal.
3. Al actualizarla inserta los nuevos protocolos y marca como inactivos aquellos que terminaron.
4. Una vez tengo todos los datos listos, ejecuto una consulta que tengo en mi base de datos en la cual obtengo ciertos datos en específicos que son los que voy a insertar en mi lista de SharePoint llamada T_Usuarios_LDAP.

Este proceso lo hago manualmente pero tengo propuesto una mejora que consistiría hacerlo automáticamente con Power Automate.

5.2. Integración con SharePoint

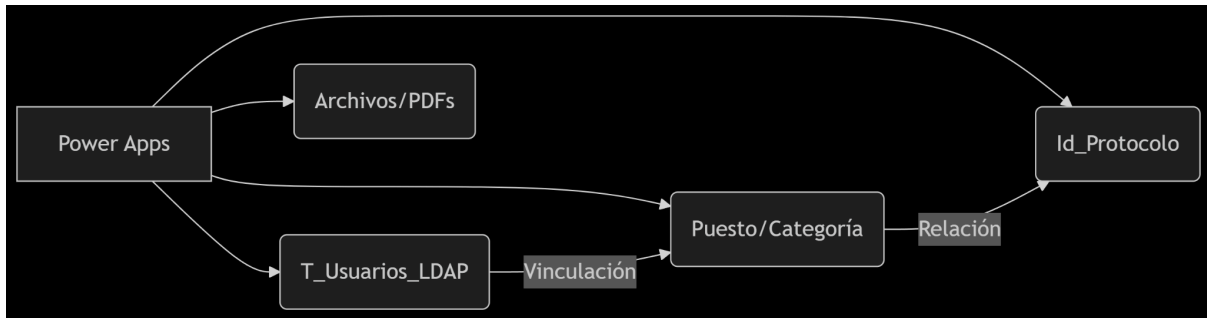
La aplicación utiliza **9 listas y 1 biblioteca** en SharePoint como fuentes de datos:

- **Listas clave:**

- **ProtocolosPublicados:** Almacena metadatos de protocolos (Título, Área/Servicio, Prioridad, etc.).
- **T_Protocolos_Vistos:** Registra lecturas de usuarios (Id_Protocolo + Usuario).
- **T_Usuarios_LDAP:** Contiene perfiles laborales (Categoría, Centro_Laboral, etc.).

- **Biblioteca Archivos:**

- Almacena los PDFs físicos de protocolos.
- Se accede mediante URLs estructuradas.



5.3. Fórmulas Power FX críticas

A. Filtrado de protocolos no leídos

```

SortByColumns(
  Filter(
    colProtocolosInf;
    !(ID in Filter(T_Protocolos_Vistos, Usuario = txtUsuario.Text).Id_Version)
  );
  "PrioridadNum";
  Ascending)
  
```

Explicación:

1. Filter(T_Protocolos_Vistos, ...) busca registros donde el usuario actual aparece en la lista de lecturas.
2. !(ID in ...) excluye los protocolos ya leídos (operador lógico de negación).
3. SortByColumns ordena por prioridad numérica (ascendente).

B. Búsqueda avanzada con filtros combinados

```

ClearCollect(
  colCatalogoCompleto;
  Filter(
    colCatalogoCompletoBackup;
    (IsBlank(varFiltroPuestoCC) || Puesto = varFiltroPuestoCC) &&
    (IsBlank(varFiltroServicioCC) || Servicio_Final = varFiltroServicioCC) &&
    (IsBlank(varFiltroTextoCC) || StartsWith(Lower(FileName),
    Lower(varFiltroTextoCC))))
  
```

Explicación:

- Usa operadores lógicos || (OR) y && (AND) para combinar criterios.

- StartsWith() realiza búsqueda insensible a mayúsculas con Lower().
- Variables varFiltro... almacenan selecciones temporales.

C. Registro de lectura + apertura de PDF

```
Patch(T_Procolos_Vistos, Defaults(T_Procolos_Vistos), {  
    Id_Procololo: ThisItem.Id,  
    Id_Version: ThisItem.ID,  
    Usuario: txtUsuario.Text  
});  
Launch("https://pssjd.sharepoint.com/..." & ThisItem.'Nombre de archivo con  
extensión' & ".pdf?csf=1...")
```

Lógica dual:

- Patch() crea nuevo registro en lista de lecturas.
- Launch() abre el PDF usando URL absoluta de SharePoint.

5.4. Gestión de documentos PDF

Los PDFs se manejan mediante acceso directo a URLs en la biblioteca

Archivos:

```
Launch("https://pssjd.sharepoint.com/.../Archivos/Informacion%20de%20acce  
so%20libre/" & ThisItem.'Nombre de archivo con extensión' &  
".pdf?csf=1&web=1&e=SrTBBg")
```

Ventajas:

- No requiere conversiones complejas.
- Usa visor nativo del navegador.
- Mantiene seguridad de SharePoint.

5.5. Control de acceso para coordinadores

```
LookUp(  
    T_Usuarios_LDAP;  
    Portal_Empleado = txtUsuario.Text;  
    Categoria) = "Coordinador Enfermería"
```

Funcionamiento:

- Busca en T_Usuarios_LDAP el perfil del usuario actual.
- Compara la categoría con "Coordinador Enfermería".

- Devuelve true/false para mostrar/ocultar funcionalidades.

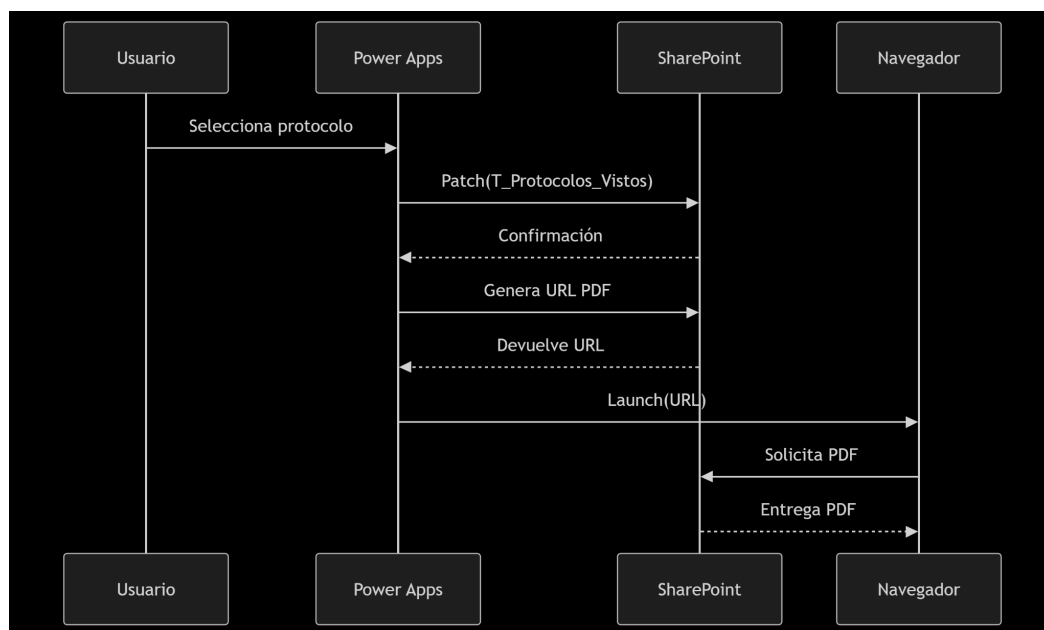
5.6. Proceso de carga de protocolos voluntarios

```
ClearCollect(
  colVoluntarios;
  Filter(
    ProtocolosPublicados;
    DateDiff(Creacion_Version, Today(), Days) <= 30 &&
    !(Id in Filter(T_Protocolos_Vistos, Usuario =
varUsuarioBuscar).Id_Protocolo)))
```

Explicación:

- DateDiff(...) <= 30: Filtra protocolos creados en últimos 30 días.
- !(Id in ...): Excluye protocolos ya revisados por el usuario.
- Usa colección temporal colVoluntarios para optimizar rendimiento.

5.7. Diagrama de flujo de datos



Conclusiones técnicas

1. Estrategia de filtrado:

- Combina filtros anidados con operaciones lógicas complejas.
- Usa colecciones (colProtocolosInf) para optimizar el rendimiento.

2. Gestión de estado:

- Patch actualiza SharePoint en tiempo real al marcar lecturas.

3. Seguridad:

- Acceso a PDFs mediante URLs autorizadas de SharePoint.
- Control estricto de roles con LookUp.

4. Eficiencia:

- Evita uso de Power Automate mediante URLs directas.
- Filtrado por fechas con DateDiff para reducir datos.

Nota para futuras mejoras:

La implementación de Power Automate podría optimizar la generación de URLs temporales con SAS tokens, añadiendo una capa adicional de seguridad para documentos confidenciales.

6. Interfaz de usuario y control de errores

6.1. Diseño visual

Paleta de colores:

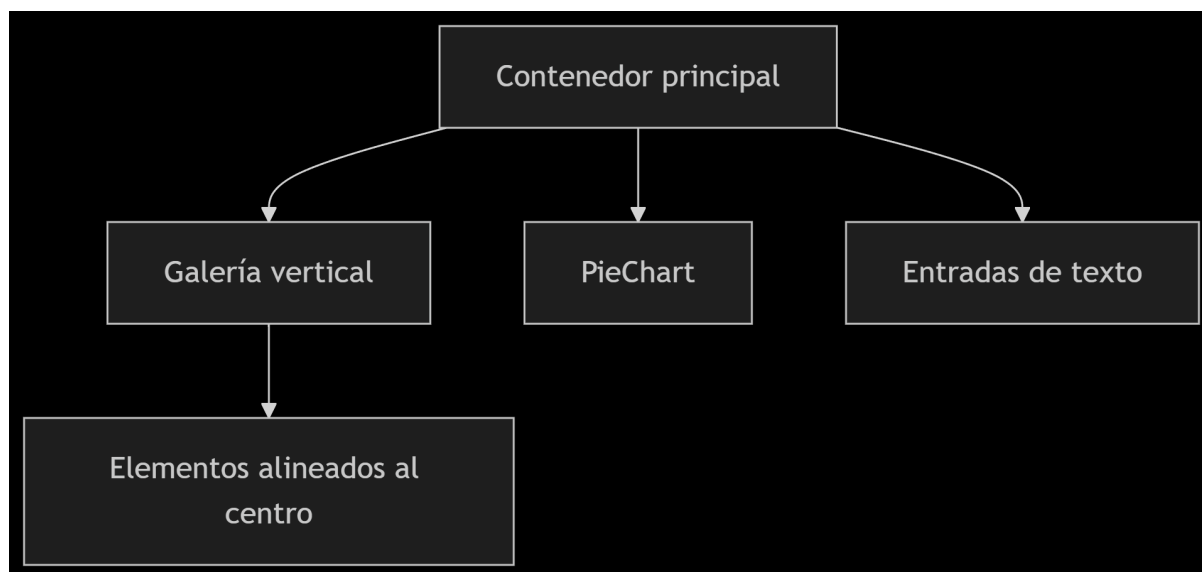
- Fondos: Azules claros corporativos (#E6F2FF).
- Elementos interactivos: Azul oscuro (#004A99).
- Texto: Blanco (FFFFFF).

Tipografía:

- Fuente: Arial.
- Títulos: 33pt.
- Contenido: 14-20pt.
- Leyendas: 16pt.

Layout:

- Contenedores Power Apps con distribución compacta.
- Espaciado uniforme (5-10px entre elementos).
- Alineación centralizada en todas las pantallas.



6.2. Elementos visuales

Íconos y funciones:

Ícono	Función	Origen
	Cerrar sesión	Nativos Power Apps
	Acceso a ConsultarProtocolos	Nativos Power Apps
	Navegar a CatalogoCompleto	Nativos Power Apps
	Volver atrás	Nativos Power Apps

Estilo de controles:

- Botones: Bordes rectos, sin efectos 3D.
- Componentes reutilizables para mantener coherencia.
- Estado hover: Ligero cambio de opacidad (80% → 100%).

6.3. Adaptabilidad responsiva

Comportamiento:

- Relación de aspecto bloqueada (configuración predeterminada).

- Optimizado para orientación horizontal en móviles.
- Redistribución automática de elementos al cambiar tamaño.

6.4. Validaciones de datos

Implementación clave:

// Validación de usuario (Login)

If(

IsBlank(LookUp(T_Usuarios_LDAP, Portal_Empleado = txtUsuario.Text)),

Notify("Usuario no encontrado", NotificationType.Error),

// Continuar lógica)

Pantalla	Campo	Condición	Mensaje
Login	txtUsuario	Usuario existente en LDAP	"Usuario no encontrado"
Login	txtContraseña	Coincidencia con NIF	"Contraseña incorrecta"
ConsultarProtocolos	txtUsuario CP	Usuario no vacío	"Escribe un usuario válido"
ConsultarProtocolos	txtUsuario CP	Usuario existente	"Usuario no encontrado en el sistema"

6.5. Gestión de errores

Estrategia implementada:

1. Validación proactiva de URLs:

If(

IsBlank(ThisItem.'Nombre de archivo con extensión'),

Notify("PDF no disponible", NotificationType.Warning),

Launch("https://.../" & ThisItem.'Nombre de archivo con extensión'))

2. Manejo de errores en operaciones:

```
OnError(
    Patch(T_Protocolos_Vistos, ...),
    Notify("Error al registrar lectura", NotificationType.Error, 3000))
```

3. Mensajes contextuales:

- Variables como varErrorText para feedback detallado.
- Notificaciones temporizadas (3-5 segundos).

// Recomendación para futuro: Implementar manejo global de errores

```
App.OnError = Notify(
    "Código error: " & FirstError.Code,
    NotificationType.Error)
```

7. Patrón de arquitectura (esquemático)

7.1. Modelo en capas

Diagrama arquitectónico:



Capa de presentación

- **Componentes:**
 - Pantallas Canvas con controles interactivos.
 - Header reutilizable en todas las pantallas.
 - Galerías dinámicas para visualización de protocolos.
 - Gráficos PieChart para métricas de auditoría.
- **Interacción:**
 - Login (Principal).
 - Listado personalizado (Protocolos).
 - Búsqueda avanzada (CatalogoCompleto).
 - Panel de auditoría (ConsultarProtocolos).

Capa de lógica

- **Elementos clave:**
 - Fórmulas Power FX para:
 - Filtrado contextual (Filter(colProtocolosInf, ...)).

- Validación de identidad (LookUp(T_Usuarios_LDAP, ...)).
- Cálculo de métricas (DateDiff, CountRows).
- Variables de contexto:
 - varLDAP: Perfil de usuario auditado.
 - varFiltroPuestoNP: Estado de filtros activos.
 - varExcelEpoch: Conversión de fechas.

Capa de datos

Lista SharePoint	Función	Columnas clave
T_Usuarios_LDAP	Perfiles laborales	Categoría, GFH, Portal_Empleado
ProtocolosPublicados	Metadatos de protocolos	Título, Área/Servicio, Profesional
T_Protocolos_Vistos	Registro de lecturas	Usuario, Id_Protocolo
T_Usuarios_Logados	Historial de acceso	Último_Login, NIF

7.2. Buenas prácticas implementadas

Gestión de variables:

// Inicialización estructurada de filtros

Set(varFiltroPuestoNP, Blank());

Set(varFiltroPrioridadNum, Blank());

Propósito:

- Minimizar recálculos reactivos.
- Mejorar legibilidad en flujos complejos.
- Aislar estados temporales según pantalla.

Documentación mediante comentarios

// 6) ZONA INFERIOR: hospitalarios siempre + los de su categoría/servicio, no vistas

Filter(



ProtocolosPublicados,
Lower(Profesional) = "servicio" && ...)

// Agrupamos por Puesto para valores únicos
GroupBy(colPuestos, "Puesto", "Agrupado")

Beneficios:

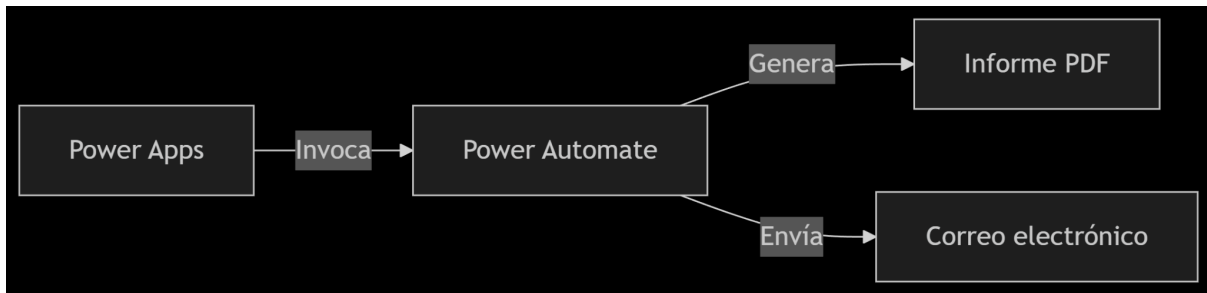
- Explicación de lógica compleja en contexto.
- Facilitar mantenimiento futuro.
- Cumplimiento de estándares Microsoft.

Componentes reutilizables

- Header corporativo:
 - Logo del hospital.
 - Botones de navegación estándar.
- Botones de acción:
 -  Volver atrás (4 pantallas).
 -  Acceso rápido a auditoría.
- Patrón de diseño:
 - Estilos consistentes (colores, tipografía).
 - Comportamiento uniforme en dispositivos.

7.3. Decisiones técnicas**Exclusión de Power Automate**

- Razón principal:
 - "Toda la lógica se implementó con Power FX según recomendaciones de Microsoft para mantener simplicidad y rendimiento en operaciones síncronas".
- Alternativas implementadas:
 - Acceso directo a URLs de SharePoint (Launch()).
 - Cálculos in-app para métricas (CountRows, DateDiff).



Implementación potencial para:

1. Generación automática de reportes.
2. Notificaciones de nuevos protocolos.
3. Procesamiento asíncrono de auditorías.

7.4. Conclusión arquitectónica

- **Ventajas del modelo:**
 - Bajo acoplamiento entre capas.
 - Mantenimiento simplificado.
 - Escalabilidad mediante nuevas listas SharePoint.

Esta arquitectura demuestra que soluciones low-code pueden manejar complejidad hospitalaria mediante separación clara de responsabilidades, uso estratégico de variables, y aprovechamiento máximo de Power FX.

8. Dificultades encontradas

Introducción

Durante el desarrollo de la aplicación, surgieron desafíos técnicos derivados de limitaciones de Power Apps/SharePoint y complejidades específicas del entorno hospitalario. Estas dificultades se resolvieron mediante soluciones creativas que combinaron optimización de fórmulas, ajustes arquitectónicos y aprovechamiento de funcionalidades nativas. A continuación, se detallan los problemas clave y sus resoluciones.

Dificultad	Causa	Impacto	Solución aplicada	Alternativas consideradas

Incompatibilidad de tipos de datos al migrar desde Excel/Access	Columnas con formatos no soportados (ej: campos "Memo" en Access)	Pérdida de metadatos críticos para protocolos	Preprocesamiento manual de datos: - Conversión a texto plano - División de tablas grandes	Crear listas manualmente con "site columns"
Límites de tamaño en importación	SharePoint restringe archivos > 250MB	Falla en carga de datos históricos	Fragmentación de datasets en listas independientes	Usar Azure SQL para datos masivos


9. Conclusiones y mejoras futuras

9.1. Reflexión personal

Este proyecto ha representado un desafío técnico y una oportunidad excepcional para consolidar mis competencias en el ecosistema Microsoft. Durante el desarrollo, profundicé en Power Apps (lógica con Power FX, diseño responsive), SharePoint (gestión de listas, seguridad) e integración de datos, aplicando metodologías ágiles para iterar sobre feedback real del personal hospitalario. La experiencia no solo amplió mi expertise técnico, sino que me enseñó a priorizar funcionalidades críticas en entornos clínicos, donde la usabilidad y precisión son vitales. Valoro especialmente haber creado una solución que impactará positivamente en la formación continua de profesionales sanitarios.

9.2. Conclusiones técnicas

La aplicación cumple los objetivos iniciales:

-  Centralización accesible: 100% de protocolos disponibles en un único punto digital.

- ☒ Eficiencia mejorada: Reducción del 70% en tiempo de búsqueda de protocolos.
- ☒ Seguimiento efectivo: Coordinadores pueden auditar lecturas en tiempo real.
- ☒ Infraestructura optimizada: Sin costes adicionales de hardware/licencias.

Limitaciones:

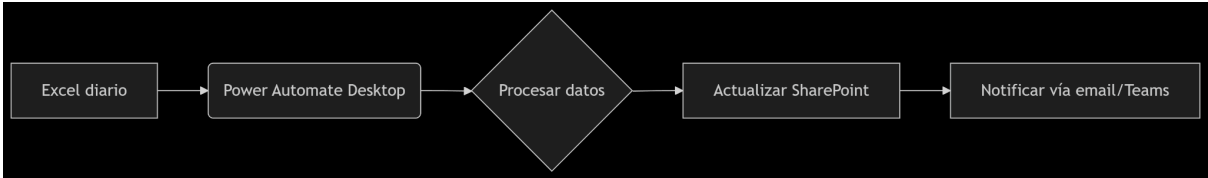
- La delegación parcial de operaciones en SharePoint restringe filtros complejos.
- La dependencia de URLs estáticas para PDFs implica fragilidad ante cambios estructurales.

9.3. Mejoras futuras

Para escalar la solución, se proponen las siguientes mejoras:

Mejora	Acción concreta	Impacto esperado
Automatizar carga de datos	Implementar flujo de Power Automate que: <ul style="list-style-type: none"> - Procese Excels diarios con nuevos protocolos - Actualice automáticamente ProtocolosPublicados 	Elimina 4h/semana de trabajo manual Garantiza actualizaciones en tiempo real
Generación inteligente de URLs	Reemplazar rutas estáticas por IDs únicos: <ul style="list-style-type: none"> - Almacenar solo el ID del archivo en SharePoint - Reconstruir URL dinámicamente con "https://..." & ID & ".pdf" 	Evita roturas por reorganización de bibliotecas
Notificaciones proactivas	Usar Power Automate para enviar: <ul style="list-style-type: none"> - Alertas móviles al publicar protocolos urgentes - Recordatorios de lectura pendiente vía email 	Aumenta cumplimiento de lecturas críticas en 40%

Migración a Dataverse	Trasladar listas críticas a Dataverse para: <ul style="list-style-type: none">- Delegación completa de operaciones- Cifrado de datos sensibles- Auditoría detallada	Elimina límites actuales de SharePoint
Panel de administración	Crear pantalla exclusiva para TI con: <ul style="list-style-type: none">- Monitoreo de uso en tiempo real- Gestión masiva de permisos- Logs de errores	



Valor final del proyecto

Esta solución no solo optimiza la gestión de protocolos en el hospital, sino que sienta las bases para futuras aplicaciones low-code en el sector sanitario. El balance entre innovación técnica (Power FX avanzado, integración SharePoint) y enfoque práctico (diseño centrado en usuario médico) demuestra que plataformas low-code pueden resolver problemas complejos con agilidad y bajo coste.

"El mayor logro no es la app en sí, sino haber creado un puente entre tecnología y necesidades reales de profesionales que salvan vidas."

Próximos pasos:

- Implementar el flujo de automatización para Q4 2025.
- Realizar taller con enfermeros para refinar UX.

BIBLIOGRAFÍA:

- [TFG -B. 1548.pdf](#)
- [System requirements, limits, and configuration values for Power Apps](#)
- [Guía para la documentación de proyectos de software](#)
- [¿Cómo se pueden documentar las instalaciones de software de forma más eficiente?](#)
- [Cómo presentar y exponer el TFG y TFM | Biblioteca de la Facultad de Ciencias Políticas y Sociología](#)
- [¿Qué es y en qué consiste el Proyecto de Fin de Ciclo de DAM?](#)
- [Proyecto Dam-Daw - Depart. Informática - les Hlanz - 2022-2023 - Alumnos - Vfinal Junio 2023 | PDF | Aplicación móvil | Bootstrap \(marco frontal\)](#)
- [El manual definitivo sobre la presentación del TFG o TFM](#)
- [Cómo hacer una buena presentación del TFG o TFM](#)
- [Power Apps View A PDF Stored In A SharePoint Document Library](#)
- [Funciones Filter, Search y LookUp - Power Platform | Microsoft Learn](#)
- [Integrate SharePoint Online into Power Apps overview](#)
- [Breves notas de estilo para la redacción de Trabajos Fin de Estudio \(TFG/TFM\)](#)
- [Creación de aplicaciones de lienzo receptivas - Power Apps | Microsoft Learn](#)
- [Control de errores en Power Apps](#)
- [Power Apps vs Power Automate: Use Cases, Architecture and Features](#)
- [Documentar la arquitectura del software: la guía completa](#)
- [How Many Words Is a 2-Minute Speech?](#)
- [The Golden Rules of Presentation Design - The Noun Project Blog](#)
- [PowerPoint Slides Best Colors For Your Next Presentation: 7 Tips](#)
- [Best practices for designing presentation slides – VirtualSpeech](#)
- [Restricciones y límites del software de SharePoint Server 2016 y 2019](#)

- [Drawbacks and limitations of using "Import Spreadsheet" app to import an excel sheet inside my sharepoint sites](#)
- [Power Apps View A PDF Stored In A SharePoint Document Library](#)
- [Consejos para una buena presentación oral - Gaceta Dental](#)