



HOSPITAL

PROYECTO PAGINA WEB HOSPITAL

JUAN JOSÉ CADAVID DÍAZ
SEBASTIÁN DUQUE SEPULVEDA
ALEJANDRO GONZALO

UNIVERSIDAD LIBRE DE PEREIRA
PROGRAMA DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SOFTWARE
RISARALDA
PEREIRA
2024



HOSPITAL

HIPERVINCULOS

RF. (Requerimiento Funcional)	Hipervínculo	RNF. (Requerimiento no Funcional)	Hipervínculo
RF01. Presentación de Información Verídica y Actualizada	RF01	RN01. Base de datos	RN01
RF02. Programa permite al usuario ingresar nuevos datos	RF02	RN02. Filtros y palabras clave	RN02
RF03 Accesibilidad a través de una Página Web Pública	RF03	RN03. Presentación de datos	RN03
RF04. permite hacer cambios en el sistema y refrescar en tiempo real	RF04	RN04. Actualizaciones constantes	RN04
RF05. Actualización en Tiempo Real	RF05	RN05. diseño amigable para el usuario	



HOSPITAL

RF06. Buscador con Indicadores Claves y Perfiles Epidemiológicos	RF06		
RF 07. Rendimiento y Escalabilidad	RF07		
RF 08. Cambios en el sistema y refrescar en tiempo real	RF08		
	RF09		
Diagrama de Componentes		Diagrama de Componentes	



HOSPITAL



HOSPITAL

REQUERIMIENTOS FUNCIONALES (RF).

RF01. Presentación de Información Verídica y Actualizada

Función	Gestión de Presentación de Información Verídica y Actualizada
Descripción	Se incluirán elementos en el escenario que podrán ser destruidos por el jugador para acceder a áreas bloqueadas y limitar el mapa.
Entradas	Datos de entrada desde fuentes de información confiables.
Fuente	Interacciones del Usuario. Base de datos de elementos del escenario.
Salidas	Estado actualizado del elemento (entero o destruido) y el indicador de si se ha desbloqueado una nueva área.
Destino	El motor de renderizado del juego, el cual mostrará si un elemento ha sido destruido, mostrará esta destrucción en pantalla. Sistema de lógica del juego, para mantener actualizado el juego.
Acción	Se verifica si corresponde a un elemento que está marcado como destructible en la base de datos del escenario, se analiza la acción realizada por el jugador (el tipo de ataque y la resistencia del elemento). Luego, se compara el estado actual (r2) con los estados previos (r1 y r0) del elemento, esto, para saber si el elemento ha resistido a ataques previos y poder determinar si la destrucción es posible. Si



HOSPITAL

	<p>la acción es válida, se actualiza el estado del elemento a destruido y se actualiza el acceso a una nueva área.</p> <p>Para elementos indestructibles, se verifica si corresponde a un elemento indestructible, se procesa la acción del jugador y se confirma que el elemento no se puede destruir, por lo que se mantiene el estado del elemento como intacto.</p>
Requerimientos	Acceso a la base de datos del escenario, dos estados previos del elemento para comparar y tomar decisiones.
Precondición	<p>El elemento debe existir en el escenario y estar en un estado interactivo.</p> <p>El jugador debe estar en una posición que permita la interacción con el elemento.</p>
Postcondición	<p>Si el elemento es destruido, r0 se sustituye con r1 y r1 se sustituye con r2 y se desbloquean nuevas áreas si es necesario.</p> <p>Si es indestructible, el estado del elemento se mantiene intacto.</p>
Efectos colaterales	<p>Si el elemento es indestructible, no tendrá acceso a ciertas áreas del mapa.</p> <p>Si es destructible, habrá cambios en las rutas lo que podrá alterar la estrategia del jugador.</p>
Diagrama de actividades	Diagrama de casos de uso



HOSPITAL

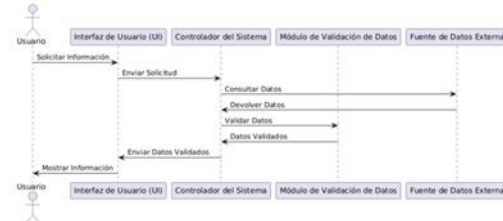
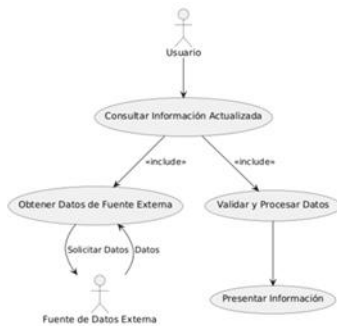


Diagrama de secuencia



Código



HOSPITAL

RF02. Programa permite al usuario ingresar nuevos datos

Función	Seleccionar escenario
Descripción	El sistema debe permitir al usuario ingresar nuevos datos a través de los formularios o interfaces correspondientes.
Entradas	El sistema debe permitir al usuario ingresar nuevos datos a través de los formularios o interfaces correspondientes.
Fuente	Base de Datos
Salidas	Los nuevos datos se guardan en la base de datos y se muestran en las secciones correspondientes del sistema.
Destino	Página web
Acción	El usuario ingresa nuevos datos, la página web los envía al sistema, la base de datos lo almacena y procesa.
Requerimientos	Base de datos activa para recibir la información y página web funcional
Precondición	El sistema debe tener formularios funcionales y conectividad con la base de datos para almacenar la información.
Postcondición	Los datos ingresados se almacenan correctamente, están disponibles para su visualización y no presentan errores en la entrada o almacenamiento.
Efectos colaterales	



HOSPITAL

Diagrama de actividades

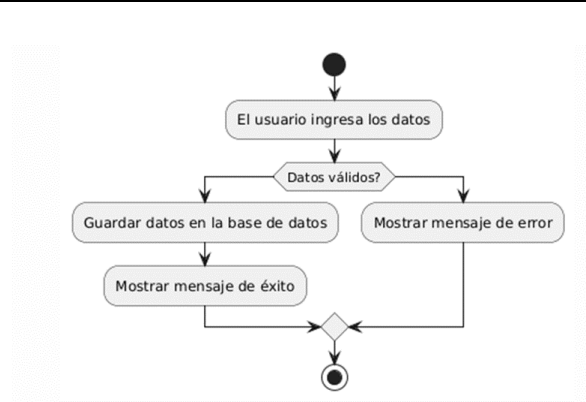
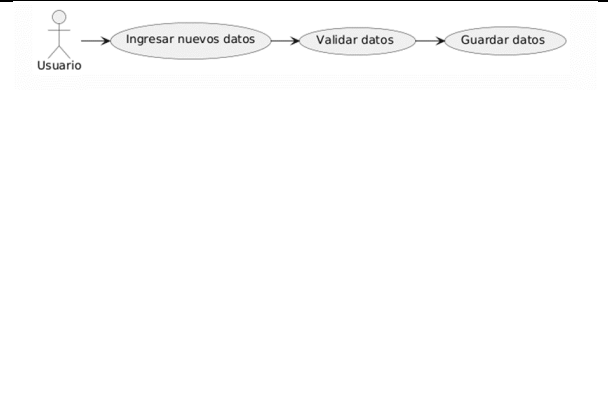
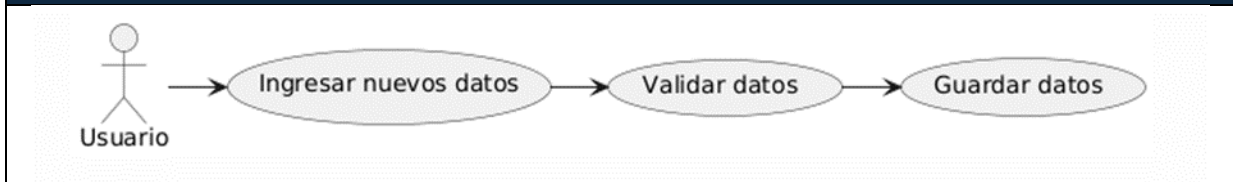


Diagrama de casos de uso



Representación

Diagrama de secuencia





HOSPITAL

RF03. Diseño amigable para el usuario

Función	Verificar compatibilidad
Descripción	El sistema debe contar con un diseño amigable e intuitivo que facilite la navegación y la interacción del usuario.
Entradas	Interacciones del usuario con la interfaz, como clics, desplazamientos y entradas en formularios.
Fuente	Página web
Salidas	Interfaz de la página web
Destino	Ciclo de control, el cual decide si el sistema debe continuar ejecutándose o si debe finalizar si no es compatible.
Acción	El sistema
Requerimientos	Acceso a la información del sistema operativo y tener conocimiento de los requisitos de compatibilidad del software.
Precondición	El sistema debe estar diseñado bajo principios de usabilidad y experiencia de usuario (UX), utilizando elementos gráficos claros y organizados.
Postcondición	Ninguna.
Efectos colaterales	Si el sistema operativo no cumple con los requisitos de compatibilidad, puede resultar en la interrupción de la ejecución del software.
Diagrama de actividades	Diagrama de casos de uso



HOSPITAL

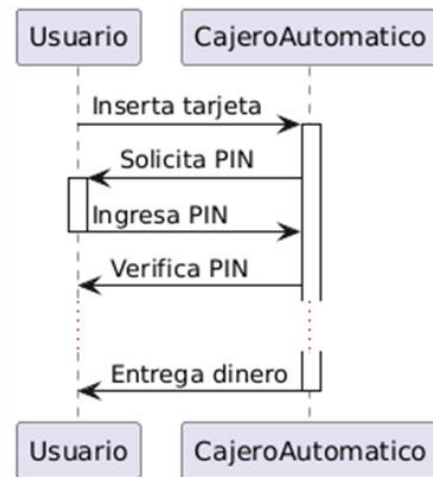
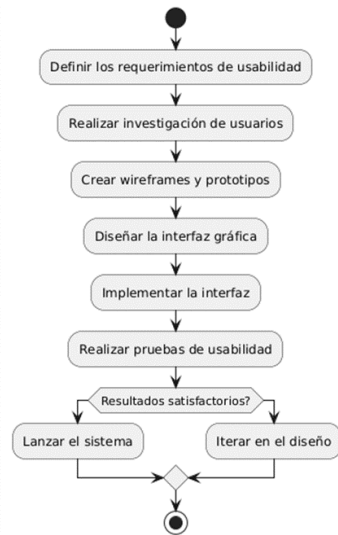
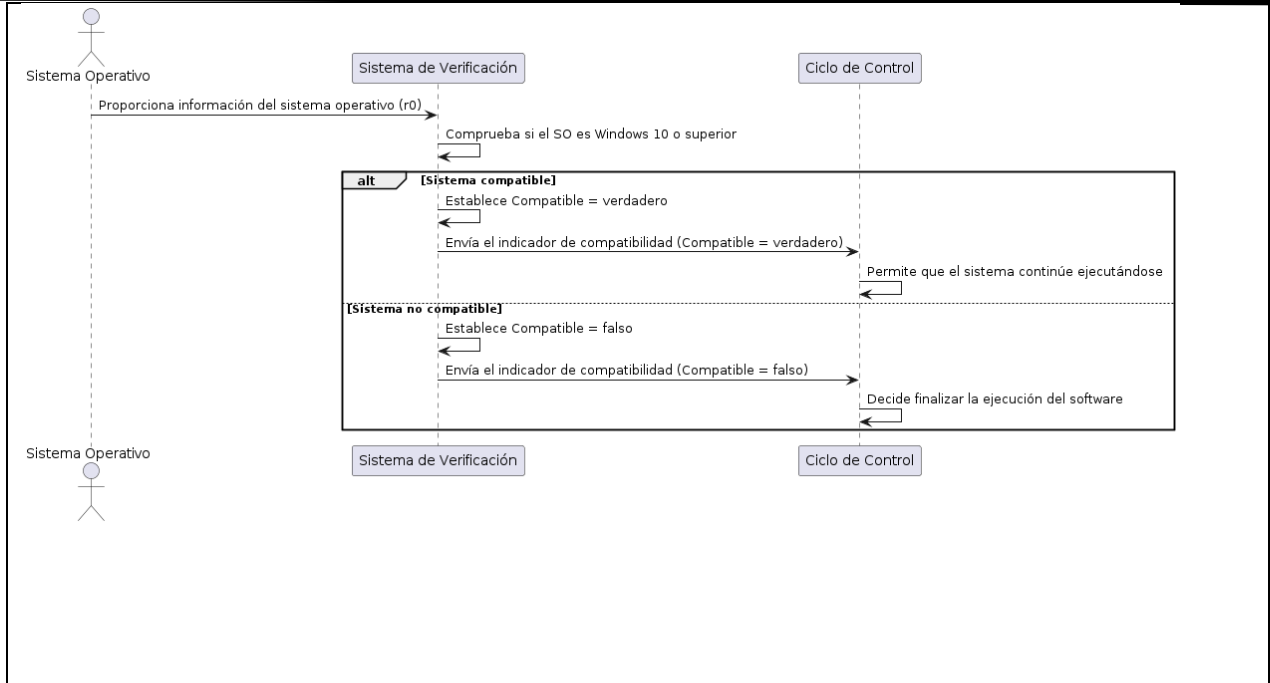


Diagrama de Secuencias



HOSPITAL





HOSPITAL

RF04 Accesibilidad a través de una Página Web Pública

Función	Definir los modos de juego disponibles.
Descripción	El sistema debe ser accesible a través de una página web pública, permitiendo a los usuarios registrados realizar consultas, realizar transacciones y acceder a la información de forma segura y eficiente.
Entradas	Solicitud de acceso a la plataforma por parte del usuario. Datos de autenticación del usuario (usuario y contraseña). Solicitudes de información o realización de transacciones.
Fuente	Sistema, base de datos
Salidas	Interfaz web intuitiva y fácil de usar. Acceso a la información solicitada de forma clara y concisa. Posibilidad de realizar transacciones en línea de manera segura. Mensajes de error claros y concisos en caso de fallos.
Destino	Página web
Acción	El sistema interactúa con la base de datos y manda la información
Requerimientos	La configuración del juego que define los modos de juego disponibles.

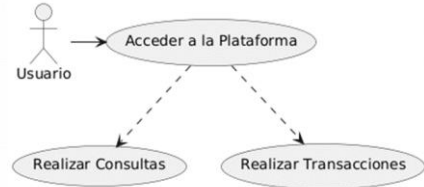
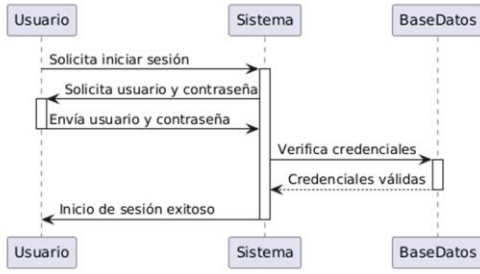


HOSPITAL

Precondición	<p>El sistema debe contar con un servidor web configurado y accesible desde internet.</p> <p>La interfaz web debe ser compatible con los navegadores más utilizados.</p> <p>Se debe implementar un mecanismo de autenticación de usuarios.</p> <p>Se deben establecer políticas de seguridad para proteger la información del usuario.</p>
Postcondición	<p>Los usuarios registrados pueden acceder a la plataforma desde cualquier dispositivo con conexión a internet.</p> <p>Las transacciones realizadas a través de la plataforma se registran de forma segura.</p> <p>La información presentada al usuario es actualizada y precisa.</p> <p>Se garantiza la confidencialidad de los datos del usuario.</p>
Efectos colaterales	Ninguno.
Diagrama de actividades	Diagrama de casos de uso y Secuencia

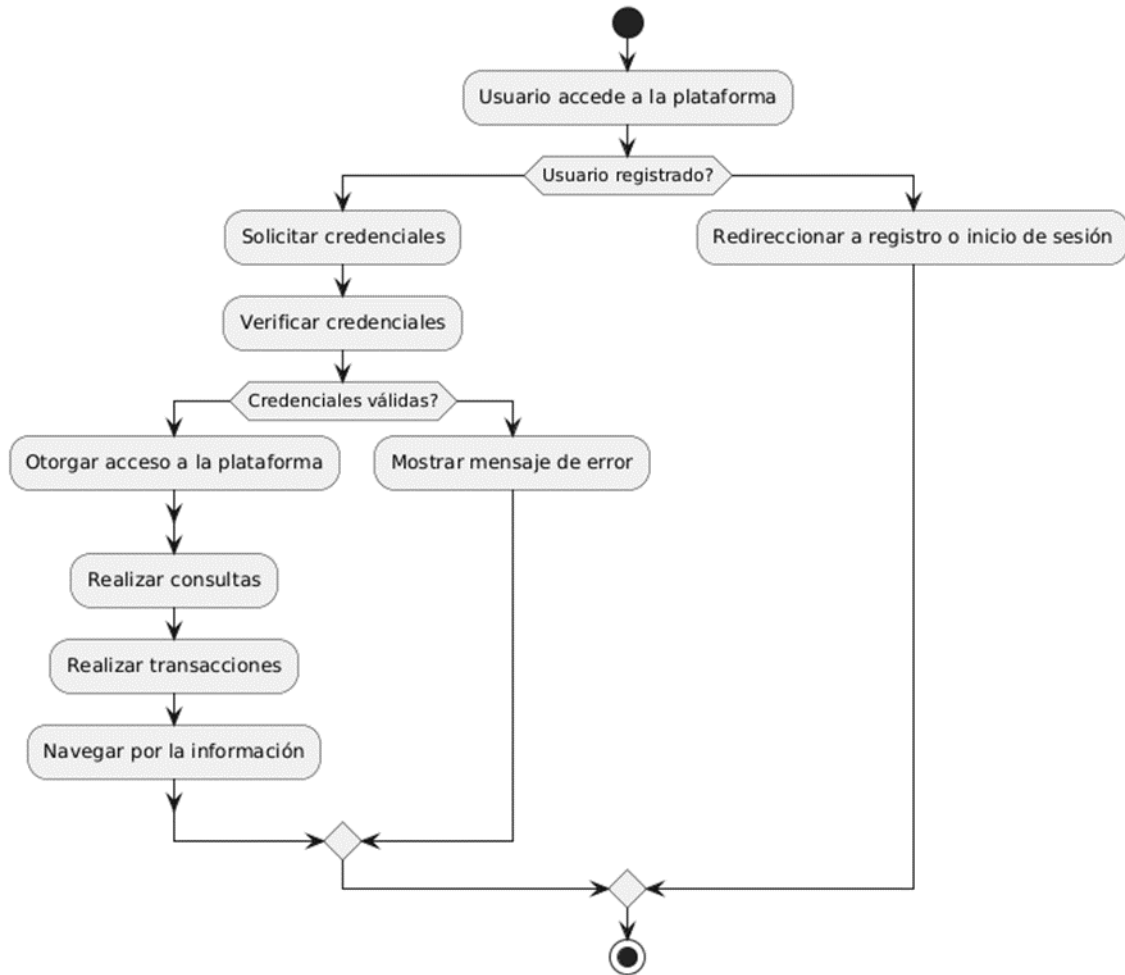


HOSPITAL





HOSPITAL





HOSPITAL

RF05. permite hacer cambios en el sistema y refrescar en tiempo real

Función	Mantenimiento y actualización continua.
Descripción	Actualizaciones constantes.
Entradas	<ul style="list-style-type: none">-Datos nuevos (contenido del juego, elementos)-Actualizaciones de software (parches, mejoras)-Retroalimentación del usuario (sugerencias, reportes de errores)
Fuente	Equipo de desarrollo, sistema de actualización.
Salidas	Sistema actualizado, mejoras continuas.
Destino	Sistema de juego, base de datos.
Acción	Verificar, implementar y mantener actualizaciones continuas.
Requerimientos	Las actualizaciones deben ser compatibles con las versiones anteriores y no deben afectar la integridad de los datos almacenados.
Precondición	El sistema debe estar en estado operativo y preparado para recibir actualizaciones.

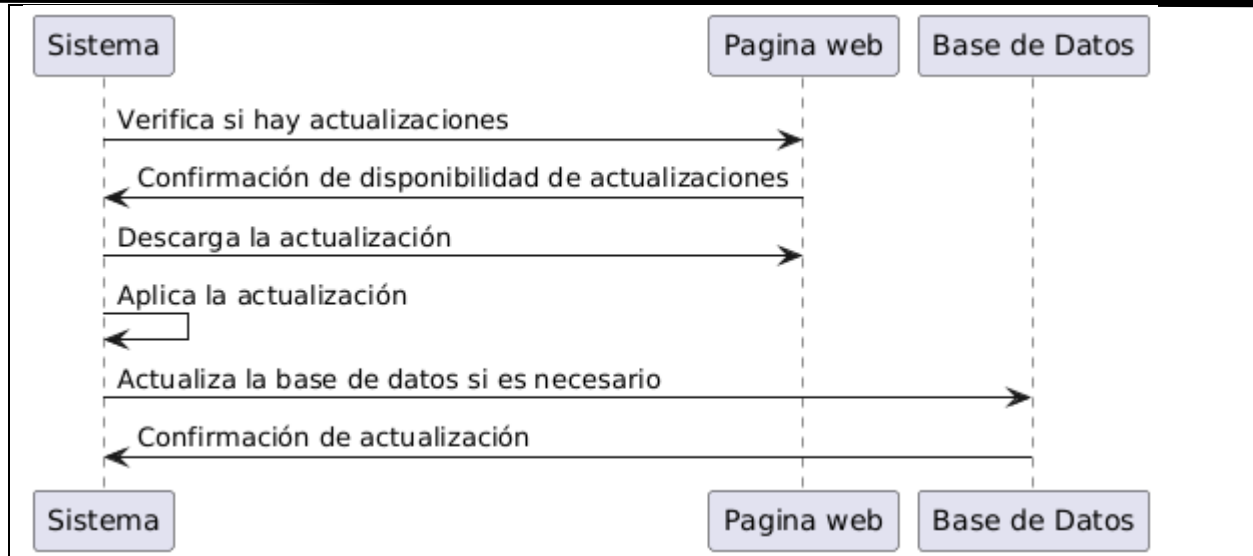


HOSPITAL

Postcondición	Las actualizaciones se aplican correctamente, y el sistema sigue siendo estable y funcional.
Efectos colaterales	N/A
Diagrama de actividades	Diagrama de casos de uso
<pre>graph TD; Start(()) --> A[El sistema verifica la disponibilidad de actualizaciones]; A --> B[Descarga las actualizaciones si están disponibles]; B --> C[Aplica las actualizaciones]; C --> D[Actualiza la base de datos si es necesario]; D --> E[Confirma la actualización]; E --> End((()));</pre>	<pre>graph TD; Actor SW[Sistema Web]; Actor SA[Servidor de Actualizaciones]; SW -- "Comprueba actualizaciones disponibles" --> UC1([Verificar Disponibilidad de Actualizaciones]); UC1 -- "Descarga si está disponible" --> UC2([Descargar Actualización]); UC2 -- "Aplica la actualización" --> UC3([Aplicar Actualización]); UC3 -- "Actualiza la base de datos si es necesario" --> UC4([Actualizar Base de Datos]); UC3 -- "Verifica que el sistema funciona correctamente" --> UC5([Verificar Integridad del Sistema]);</pre>
Diagrama de secuencia	



HOSPITAL

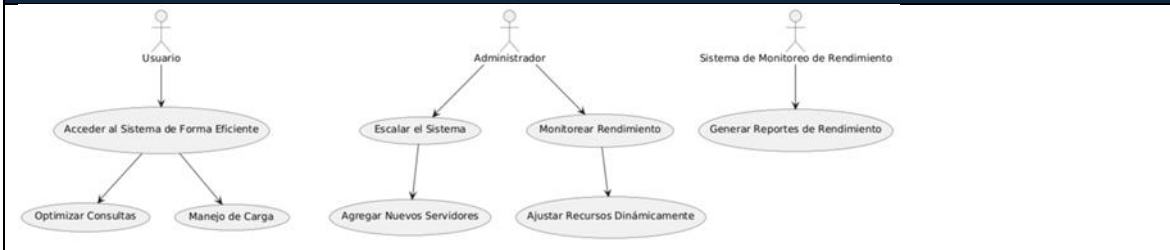




HOSPITAL

RF06. Actualización en Tiempo Real

Función	Mantener la página web actualizada
Descripción	El sistema debe actualizarse en tiempo real, reflejando cualquier cambio o nueva información sin necesidad de que el usuario realice acciones
Entradas	Solicitudes del sistema
Fuente	Sistema, página web
Salidas	Respuesta del sistema en el menor tiempo posible bajo carga normal.
Destino	Base de datos, página web
Acción	Inicia el programa (r0) carga la información (r1) y muestra el programa (r2).
Precondición	Arquitectura del sistema diseñada para ser escalable
Postcondición	La interfaz se actualiza dinámicamente para reflejar la información más reciente. Según actualización
Efectos colaterales	N/A
Diagrama de actividades	Diagrama de casos de uso





HOSPITAL

RF07. Buscador con Indicadores Claves y Perfiles Epidemiológicos

Función	Buscador el cual busca en la base de datos claves específicas de perfiles epidemiológicos
Descripción	El sistema debe incluir un buscador que permita a los usuarios encontrar información utilizando indicadores claves, perfiles epidemiológicos o temas de interés.
Entradas	Datos entrantes de fuentes en tiempo real. O por tiempos dependiendo del sistema que se use como base de datos Api
Fuente	Sistema y base de datos
Salidas	Lista de resultados relevantes y filtrados basados en los términos de búsqueda introducidos.
Destino	Página web
Acción	El usuario ingresa la información requerida, la página le solicita al sistema la información precisa,
Requerimientos	RN06 (Recursos externos), Animaciones
Precondición	Existencia de un índice de búsqueda optimizado con los datos disponibles.
Postcondición	El usuario recibe una lista filtrada de resultados correspondientes a su consulta.



HOSPITAL

Efectos colaterales

N/A

Diagrama de actividades

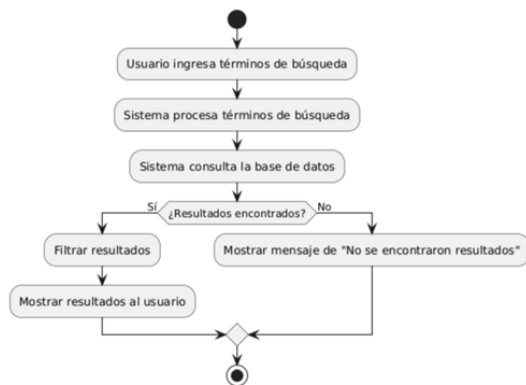


Diagrama de casos de uso



Diagrama de secuencia





HOSPITAL

RF 08. Rendimiento y Escalabilidad

Función	Velocidad y estabilidad a la hora de usar el sistema
Descripción	El sistema debe mantener un rendimiento óptimo y ser escalable para manejar un número creciente de usuarios y datos sin degradación del rendimiento.
Entradas	Solicitudes de usuarios concurrentes.
Fuentes	Entrada del usuario a través del teclado o controlador de juego, configuraciones guardadas.
Salidas	Respuesta del sistema en el menor tiempo posible bajo carga normal.
Acción	El sistema mantiene su condición estable a la hora de su uso
Destino	Sistema, base de datos y página web
Precondición	Arquitectura del sistema diseñada para ser escalable



HOSPITAL

Postcondición

El sistema responde de manera eficiente incluso con un alto número de usuarios simultáneos.

Efectos colaterales

Cambio en el estado del juego (posición del jugador, estado de los enemigos, estado de las bombas).

Diagrama de actividades

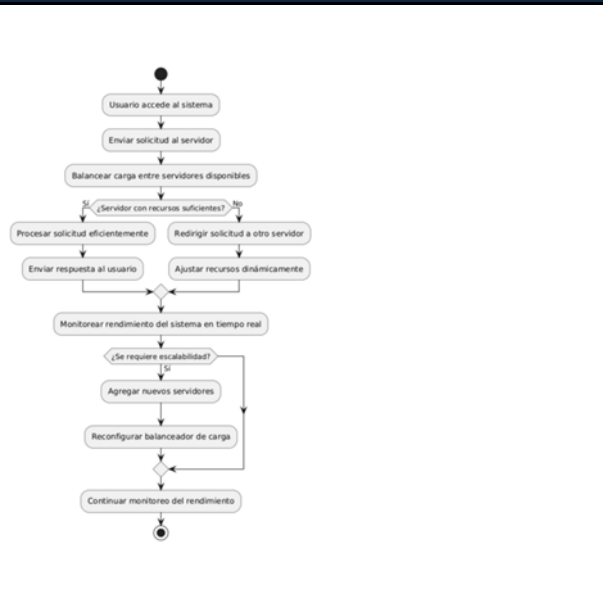


Diagrama de casos de uso

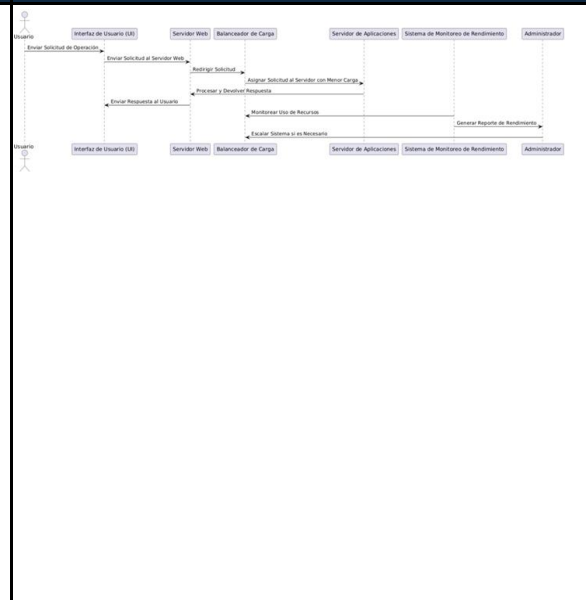
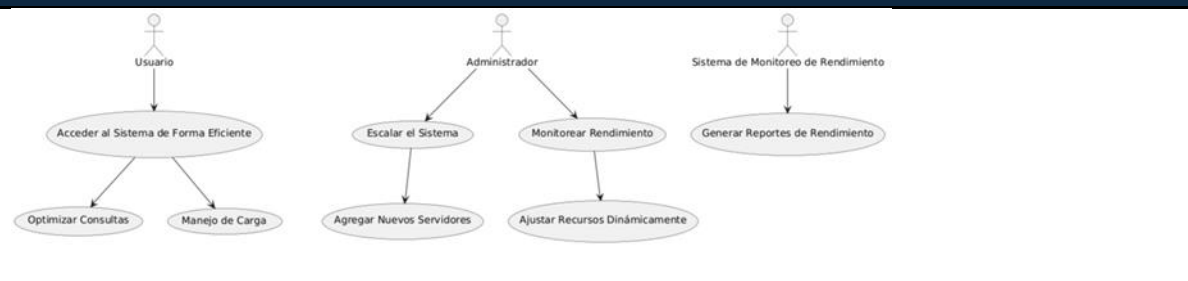


Diagrama de secuencia





HOSPITAL



HOSPITAL

RF 09. Cambios en el sistema y refrescar en tiempo real

Función	Cambios en el sistema y refrescar en tiempo real en la página web.
Descripción	El sistema debe permitir realizar cambios en el sistema y reflejarlos en tiempo real sin necesidad de recargar la página.
Entradas	Cambios realizados por el usuario en el sistema (como modificaciones en el contenido, ajustes de configuración, etc.).
Fuentes	Entradas del usuario a través de la página web, página web
Salidas	Actualización inmediata de los datos visualizados en el sistema sin recargar la página.
Acción	El sistema de juego procesa las entradas del jugador y actualiza el estado del juego.
Destino	Página web.
Precondición	El sistema debe estar implementado utilizando tecnologías de



HOSPITAL

	actualización en tiempo real, como WebSockets o AJAX.
Postcondición	El sistema debe estar implementado utilizando tecnologías de actualización en tiempo real, como WebSockets o AJAX.
Efectos colaterales	Ninguno.
Diagrama de actividades	Diagrama de casos de uso
<pre>graph TD Start(()) --> A[Usuario realiza cambio en la UI] A --> B[Enviar solicitud al servidor] B --> C{¿Datos requieren actualización?} C -- Sí --> D[Procesar solicitud en el servidor] D --> E[Actualizar datos en la base de datos] E --> F{ } C -- No --> F F --> G[Enviar respuesta al cliente] G --> H[Actualizar la interfaz en tiempo real] H --> End((()))</pre>	<pre>sequenceDiagram actor User participant UI participant Server participant DB[Base de Datos] User->>UI: Realiza cambio UI->>Server: Enviar solicitud de cambio Server->>DB: Actualizar datos (si es necesario) DB-->>Server: Confirmar actualización Server-->>UI: Respuesta con actualización UI-->>User: Refrescar vista en tiempo real</pre>
Diagrama de secuencia	



HOSPITAL





HOSPITAL

REQUERIMIENTOS NO FUNCIONALES (RN).

RN01. Base de datos

Función	Gestión de la base de datos.
Descripción	Base de datos que almacena la información.
Entradas	Datos de la página web (elementos, información, etc.) (r1) Acciones del usuario (Búsquedas, interacciones, etc.) (r2) Solicitud al sistema para buscar la información (r3) Enviar la información a la página para mostrarla al usuario (r4)
Salidas	Interacciones del usuario, sistema de registro del programa. (r5)
Destino	Página web. Usa la información que le proporcione la base de datos
Acción	Usuario realiza una acción en la página web y la pagina web crea una señal de carga (r1), la página web envía la información a la base de datos (r2) la base de datos busca la solicitud con ayuda del sistema (r3) la base de datos obtiene la solicitud (r5) la página muestra la solicitud
Requerimientos	La base de datos debe manejar las entradas que la página web requiera.



HOSPITAL

Función	Gestión de la base de datos.
Descripción	Base de datos que almacena la información.
Entradas	Datos de la página web (elementos, información, etc.) (r1) Acciones del usuario (Búsquedas, interacciones, etc.) (r2) Solicitud al sistema para buscar la información (r3) Enviar la información a la página para mostrarla al usuario (r4)
Precondición	La base de datos debe estar conectada y operativa.
Postcondición	Los datos requeridos se deben enviar al usuario.
Efectos colaterales	Ninguno
Diagrama de actividades	Diagrama de casos de uso y Secuencia



HOSPITAL

Función	Gestión de la base de datos.
Descripción	Base de datos que almacena la información.
Entradas	<p>Datos de la página web (elementos, información, etc.) (r1)</p> <p>Acciones del usuario (Búsquedas, interacciones, etc.) (r2)</p> <p>Solicitud al sistema para buscar la información (r3)</p> <p>Enviar la información a la página para mostrarla al usuario (r4)</p>
<pre> graph TD Start(()) --> A[Usuario interactúa con la pagina] A --> B[El sistema registra la interacción] B --> C[El sistema envía los datos a la base de datos] C --> D{Los datos son válidos?} D -- Si --> E[La base de datos almacena los datos] D -- No --> F[Error en la verificación] E --> G{ } F --> G G --> H[El sistema envía la información a la página] H --> End(()) </pre>	<pre> sequenceDiagram actor Usuario participant Pagina web participant Base de datos Usuario->>Pagina web: Solicita reporte o visualización de datos Pagina web->>Base de datos: Consulta la base de datos Base de datos-->>Pagina web: Devuelve los datos solicitados Pagina web-->>Usuario: Muestra los datos según las preferencias </pre>



HOSPITAL

RN02. Filtros y palabras clave

Función	Optimización de la búsqueda
Descripción	Filtros y palabras clave para el buscador.
Entradas	Palabras clave ingresadas por el usuario (r1) Parámetros de filtro (fechas, categorías, estados) (r2) Configuración del usuario (preferencias de búsqueda) (r3)
Fuente	Interacciones del usuario con el buscador.
Salidas	Resultados de búsqueda filtrados y relevantes. (r4)
Destino	Página web, muestra la información relevante al usuario.
Acción	Usuario realiza una búsqueda en el buscador (r1), El buscador saca las palabras claves, fechas, etc. y la página web muestra una señal de carga (r2) la página web realiza la solicitud (r3) la base de datos devuelve la solicitud (r4) la página web muestra la solicitud
Requerimientos	El sistema debe procesar y devolver resultados en tiempo real.
Precondición	El usuario debe haber ingresado una palabra clave o un filtro para iniciar la búsqueda.

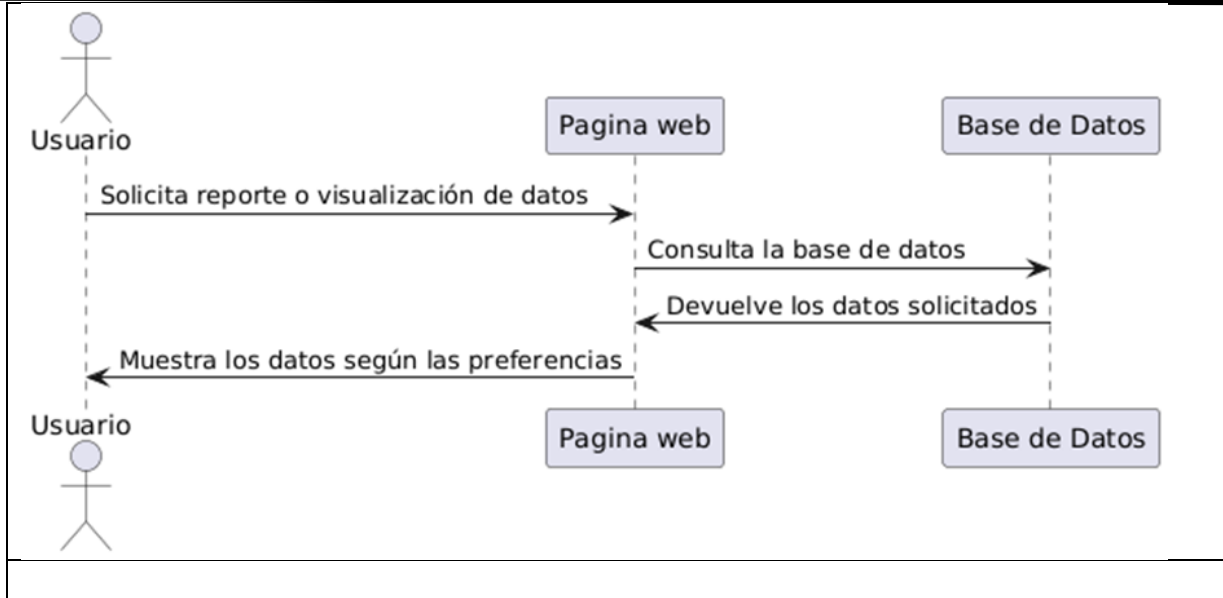


HOSPITAL

Postcondición	Los resultados mostrados deben ser relevantes y respetar los filtros y parámetros definidos.
Efectos colaterales	
Diagrama de actividades	Diagrama de casos de uso y Secuencia
<pre> graph TD Start(()) --> A[Usuario ingresa palabras clave] A --> B[El buscador procesa los filtros y las palabras clave] B --> C[El buscador consulta la base de datos] C --> D[La base de datos devuelve los resultados] D --> E[El buscador presenta los resultados al usuario] E --> End((())) </pre>	<pre> sequenceDiagram actor User participant System participant DB as Base de Datos participant Admin as Administrador User->>System: El usuario ingresa palabras clave, filtros, etc. activate System System->>DB: El sistema consulta la base de datos activate DB DB-->>System: La base de datos devuelve los resultados deactivate DB System->>User: El sistema presenta los resultados al usuario deactivate System User->>System: El usuario ingresa palabras clave, filtros, etc. activate System System->>DB: El sistema consulta la base de datos activate DB DB-->>System: La base de datos devuelve los resultados deactivate DB System->>User: El sistema presenta los resultados al usuario deactivate System User->>System: El usuario ingresa palabras clave, filtros, etc. activate System System->>DB: El sistema consulta la base de datos activate DB DB-->>System: La base de datos devuelve los resultados deactivate DB System->>User: El sistema presenta los resultados al usuario deactivate System </pre>



HOSPITAL





HOSPITAL

RN03. Presentación de datos

Función	Claridad en la presentación de datos.
Descripción	Presentación de datos clara (reportes, tablas, etc.)
Entradas	Datos procesados y almacenados (r1) Parámetros de presentación (formatos, estilos) (r2) Preferencias del usuario (idioma, disposición) (r3)
Fuente	Sistema de base de datos, registros del juego.
Salidas	Reportes, tablas y visualizaciones claras. (r4)
Destino	Página web
Acción	Generar y presentar los datos de manera clara y accesible.
Requerimientos	Implementación del Presentación de datos, capacidad para gestionar las selecciones del jugador y actualizar la configuración del juego.
Precondición	Los datos deben estar completamente procesados y formateados según las preferencias del usuario.

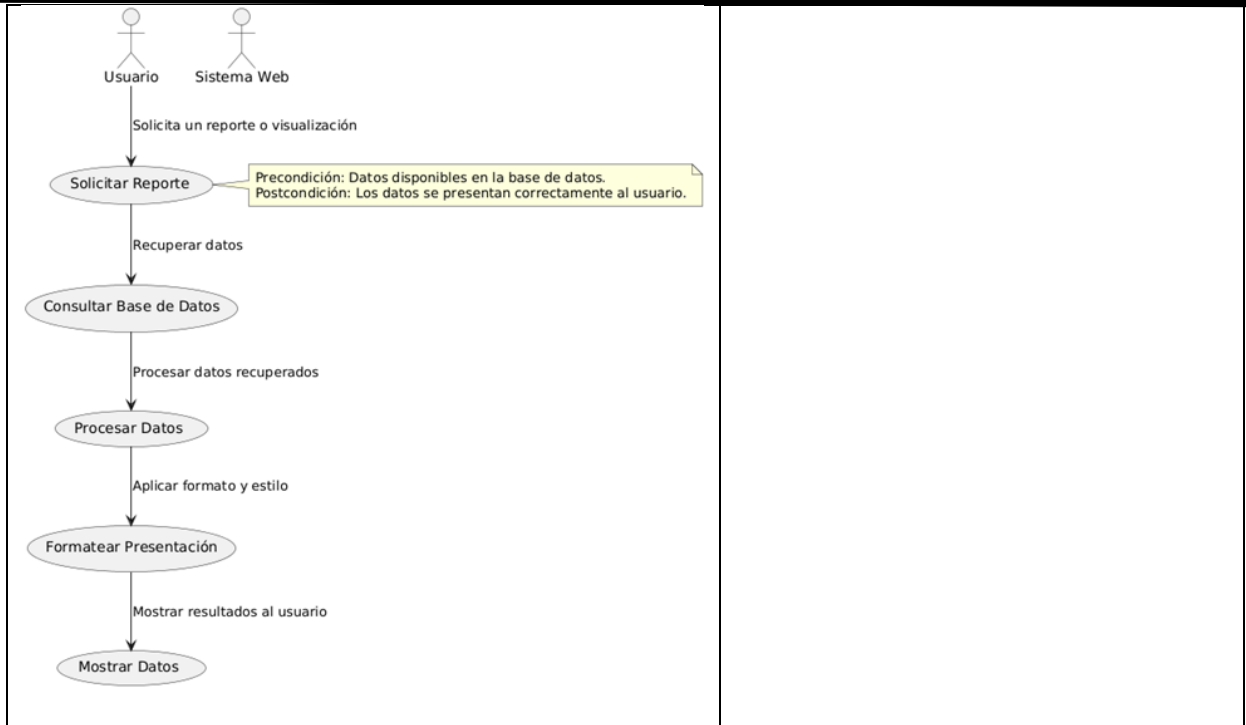


HOSPITAL

Postcondición	La presentación de datos es precisa y se ajusta a los parámetros definidos.
Efectos colaterales	
Diagrama de actividades	Diagrama de casos de uso y Secuencia
<pre>graph TD; Start(()) --> A[Usuario ingresa palabras clave]; A --> B[El buscador procesa los filtros y las palabras clave]; B --> C[El buscador consulta la base de datos]; C --> D[La base de datos devuelve los resultados]; D --> E[El buscador presenta los resultados al usuario]; E --> End((()))</pre>	<pre>sequenceDiagram; actor Usuario; participant Pagina web; participant Base de Datos; Usuario->>Pagina web: Solicita reporte o visualización de datos; Pagina web->>Base de Datos: Consulta la base de datos; Base de Datos-->>Pagina web: Devuelve los datos solicitados; Pagina web-->>Usuario: Muestra los datos según las preferencias;</pre>



HOSPITAL





HOSPITAL

RN04. Actualizaciones constantes

Función	Mantenimiento y actualización continua.
Descripción	El jugador ganará el nivel al eliminar a todos los enemigos, lo que desencadenará una pantalla de victoria.
Entradas	Cantidad de enemigos restantes (r0).
Fuente	Sistema de gestión de enemigos.
Salidas	Actualizaciones constantes cumplidas (r1).
Destino	Utilizar el resultado de la verificación de Actualizaciones constantes para determinar el flujo del juego.
Acción	Evaluar si la cantidad de enemigos restantes es igual a cero, si todos los enemigos han sido eliminados, se establece Actualizaciones constantes cumplidas (r1) en verdadero.
Requerimientos	Implementación del sistema de gestión de enemigos.
Precondición	El juego está en curso y hay enemigos presentes en el nivel.
Postcondición	Se determina si el jugador ha las Actualizaciones constantes al eliminar a todos los enemigos.
Efectos colaterales	Activación de eventos relacionados con la victoria (mensaje en pantalla).



HOSPITAL

Diagrama de actividades y Secuencia

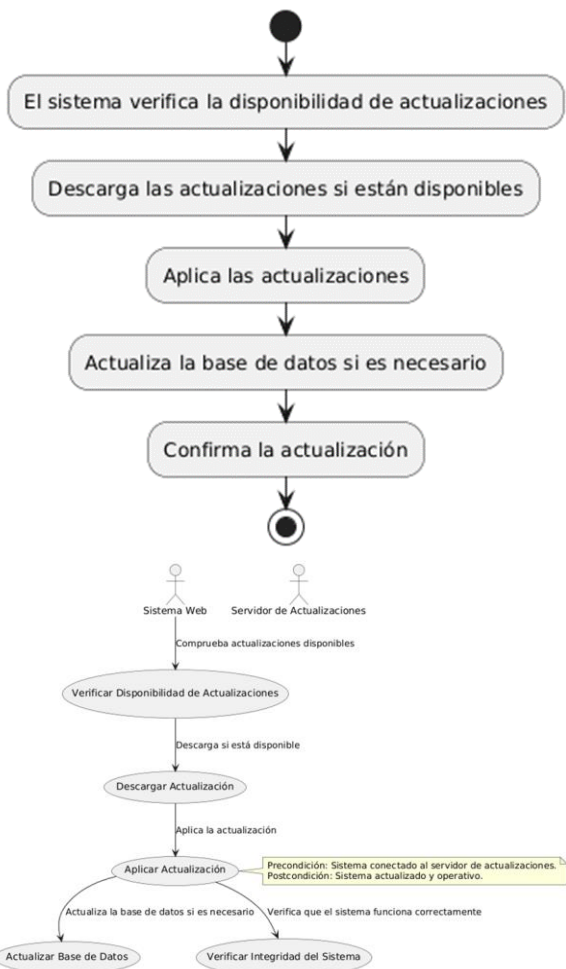
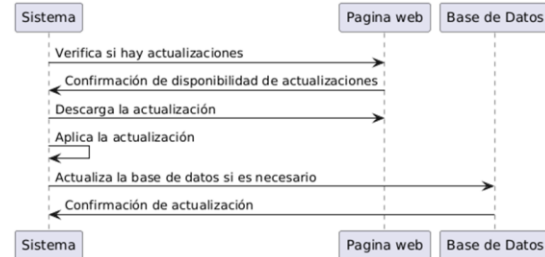


Diagrama de casos de uso





HOSPITAL



HOSPITAL



HOSPITAL



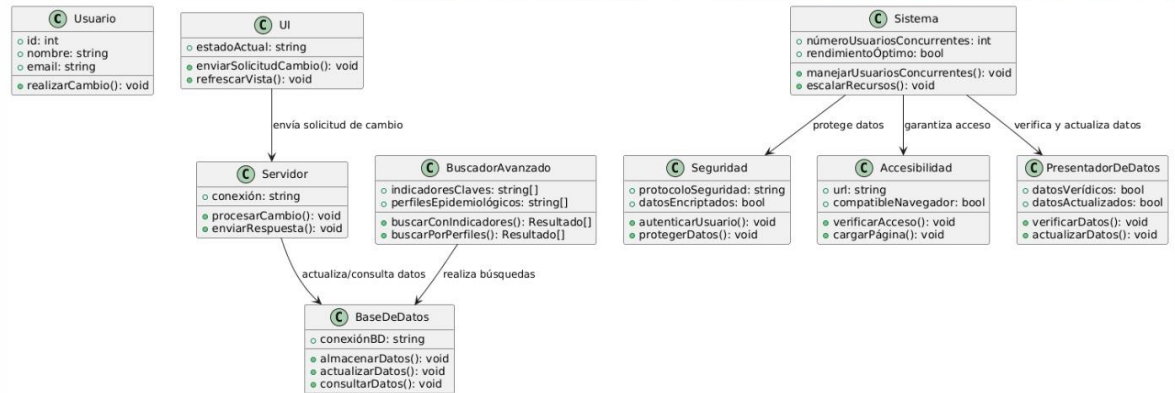
```

classDiagram
    class Usuario {
        id: int
        nombre: string
        email: string
        +realizarCambio() void
    }
    class UI {
        +estadoCambio: string
        +email: string
        +realizarCambio() void
    }
    class Servidor {
        +conexion: string
        +procesarCambio() void
        +emitarRequest() void
    }
    class Bus {
        +indicadoresCirc: string
        +identificarPedido: string
        +buscarControl: string
        +buscarPorPerfil: string
    }
    class BaseDeDatos {
        +conexionBD: string
        +emitirCambio() void
        +actualizarDatos() void
        +consultarDatos() void
    }

    Usuario --> UI
    UI --> Servidor : envia solicitud de cambio
    Servidor --> BaseDeDatos : actualiza/comunika datos
    BaseDeDatos --> Bus : recupera
    Bus --> BaseDeDatos : recupera
  
```



HOSPITAL





Muestra	



HOSPITAL





HOSPITAL