**Grado Superior**

**Desarrollo de aplicaciones web**

LifeFlow



**Plan de Empresa**

Malena Muñoz López

Aníbal Martín

**Curso 2024/25**

**Contenido**

1. **Agradecimientos y dedicatorias**
2. **Resumen**
3. **Abstract**
4. **Introducción**
5. **Alcance funcional del sistema**
6. **Diseño técnico** 
   1. Diagrama de clases
   2. Esquema de bases de datos
   3. Diseño de la interfaz de usuario
7. **Tecnología utilizada**
8. **Planificación**
9. **Presupuesto**
10. **Manual de instalación**
11. **Manual de usuario**
12. **Conclusiones**
13. **Bibliografía**
14. Logotipo

    Descripción generada automáticamente con confianza baja**Anexo**
15. **Agradecimiento y dedicatorias**

Para empezar, me gustaría expresar mi sincero agradecimiento tanto al centro como al profesorado por haber contribuido de forma clave a mi crecimiento académico y profesional. En primer lugar, su apoyo fue fundamental durante mi formación en desarrollo multiplataforma, sentando las bases de mis conocimientos en programación y estructura de software. Posteriormente, me brindaron las herramientas necesarias para expandirme hacia el ámbito web, perfeccionando mis competencias como desarrollador fullstack. Gracias a todo ello, hoy cuento con una base sólida y versátil que me permitirá afrontar con seguridad los desafíos de mi futura trayectoria laboral.

También queremos agradecer a todos los compañeros que me han acompañado durante mi año en el ciclo formativo, por esos momentos de compañerismo, apoyo mutuo y buen humor. Gracias por estar presentes en los días más exigentes, por resolver dudas juntos, por compartir ideas y, sobre todo, por hacer este camino mucho más llevadero y enriquecedor.

Quiero agradecer especialmente a todas mis amigas y a mi familia, quienes han estado presentes desde el inicio hasta la finalización de este proyecto. Su apoyo, implicación y disposición para ayudarme a recopilar información han sido fundamentales para lograr desarrollar la mejor versión posible de esta aplicación. Su acompañamiento constante ha marcado una gran diferencia en este proceso, y por ello les estaré siempre agradecida.

Finamente, quiero agradecer sinceramente a las personas que se encargarán de evaluar mi trabajo, por dedicar su tiempo y atención a este proyecto que he desarrollado con esfuerzo, constancia e ilusión. Valoro profundamente la oportunidad de poder compartir el resultado de todo este proceso, y confío en que la presentación refleje el compromiso y la dedicación que he puesto en cada etapa del desarrollo.

1. **Resumen**

LifeFlow es el trabajo de fin de curso desarrollado a partir del conocimiento adquirido durante el curso y de forma autodidacta, reflejado en una aplicación para el seguimiento de los ciclos menstruales femeninos.

Este proyecto está desarrollado por 2 partes, la parte front-end y la parte back-end. La parte front-end está desarrollada principalmente con la librería React, junto con las tecnologías básicas para el desarrollo web HTML, CSS (preprocesado de SCSS) y JavaScript. La parte front-end está conectada a la parte back-end mediante una API desarrollada desde cero. Se trata de una API desarrollada en .NET, que funciona mediante la arquitectura MVC para ofrecer mayor comodidad a la hora de consumirla.

El objetivo principal de esta aplicación es proporcionar una herramienta útil y funcional para el seguimiento del ciclo menstrual, permitiendo a las usuarias registrar y consultar información relacionada con su salud menstrual de manera sencilla y centralizada.

En cuanto al enfoque, LifeFlow nace con la intención de ofrecer una alternativa web a las aplicaciones móviles existentes. En la actualidad, este tipo de herramientas están orientadas casi exclusivamente a dispositivos móviles, lo que limita su acceso a quienes no disponen de uno o prefieren utilizar ordenadores o portátiles. Con esta propuesta se busca ampliar el alcance del servicio, favoreciendo una mayor accesibilidad y adaptándose a distintos perfiles de usuarias que demandan flexibilidad en los dispositivos que utilizan.

En resumen, LifeFlow es un trabajo en el que quiero mostrar todo lo que he aprendido académicamente y laboralmente, que refleja mis habilidades en diseño de interfaz, en mi capacidad de resolución.

1. **Abstract**

LifeFlow is the final year project developed based on the knowledge acquired both during the course and through self-directed learning, embodied in an application for tracking the female menstrual cycle.

This project is structured in two main parts: the front-end and the back-end. The front-end has been developed primarily using the React library, along with core web development technologies such as HTML, CSS (preprocessed from SCSS), and JavaScript. It is connected to the back-end through a custom-built API. This API was developed in .NET and follows the MVC architecture, offering greater ease of use and integration.

The main objective of this application is to provide a useful and functional tool for tracking the menstrual cycle, allowing users to log and review information related to their menstrual health in a simple and centralized way.

In terms of its focus, LifeFlow was created with the intention of offering a web-based alternative to existing mobile applications. Currently, most tools in this category are designed exclusively for mobile devices, which limits access for those who do not own one or prefer to use desktop or laptop computers. This proposal aims to expand service availability, improving accessibility and adapting to different user profiles who require flexibility across devices.

In summary, LifeFlow is a project through which I aim to demonstrate everything I have learned academically and professionally. It showcases my interface design skills and my ability to solve problems efficiently.

1. **Introducción**

El trabajo final de grado tiene como objetivo principal mostrar todos los conocimientos adquiridos durante el curso y la vida laboral, aplicados al desarrollo de esta aplicación. Junto con el tiempo invertido en la investigación de herramientas y la resolución de errores, se ha obtenido como resultado una web completamente funcional y usable, accesible para cualquier usuario.

LifeFlow es una aplicación web orientada al seguimiento y control de los ciclos menstruales femeninos, la planificación de eventos relacionados y la generación de reportes personalizados a partir de estadísticas. Ideada desde una perspectiva femenina y desarrollada con conocimientos tanto en programación como en la temática tratada, esta herramienta nace con el propósito de ofrecer una solución útil, intuitiva y adaptada a las necesidades reales de sus usuarios.

Ha sido diseñada con una interfaz sencilla e intuitiva, accesible desde cualquier navegador, lo que permite que cualquier persona interesada en llevar un control de su salud menstrual pueda utilizarla fácilmente.

El inicio de sesión con Google permite al usuario acceder de forma rápida y segura sin crear una cuenta adicional. Gracias a este sistema de autenticación externa, la información personal se mantiene protegida y se mejora la fluidez de uso. Además, permite integrar Google Calendar para guardar automáticamente eventos relacionados con el ciclo menstrual.

El usuario tendrá acceso a un calendario trimestral e interactivo, podiendo agregar información diaria en cada día, como notas, síntomas, emociones, pruebas de embarazo, etc. Esta función se ubica en la navegación principal ubicada en la parte superior de la página, donde se encuentran otras funciones como la creación y administración de eventos guardados en Google Calendar, y la generación de reportes basados en la información del usuario.

1. **Alcance funcional del sistema**

LifeFlow es una aplicación web diseñada para facilitar el seguimiento del ciclo menstrual de manera práctica, clara y accesible desde cualquier dispositivo con conexión a internet. Su propósito es ofrecer una herramienta funcional y centralizada para registrar y consultar información relacionada con los ciclos menstruales.

La aplicación permite a cualquier persona llevar un control detallado de su ciclo menstrual mediante el registro del inicio y la duración de cada periodo. Con estos datos, el sistema es capaz de calcular y predecir automáticamente la fecha aproximada del siguiente ciclo, así como los días estimados de fertilidad y ovulación.

LifeFlow ofrece una visualización en forma de calendario interactivo, en el que se destacan los diferentes eventos del ciclo de forma clara. Esto permite consultar tanto el estado actual como la evolución a lo largo del tiempo. Además, se incluye un historial de ciclos anteriores, útil para el análisis personal y el seguimiento continuo.

El sistema permite crear una cuenta mediante la sesión con Google para guardar los datos de forma segura, garantizando la privacidad y accesibilidad desde diferentes dispositivos. También ofrece la opción de activar notificaciones o recordatorios personalizados, con el fin de ayudar a mantener una continuidad en el registro o anticiparse a futuros eventos del ciclo, como citas médicas.

Una característica distintiva de LifeFlow es su enfoque en el entorno web, lo que la hace accesible desde ordenadores, portátiles o cualquier dispositivo con navegador, a diferencia de otras aplicaciones similares que están orientadas exclusivamente al uso móvil.

En resumen, el alcance funcional de LifeFlow incluye:

* Registro manual del inicio y duración del ciclo menstrual.
* Cálculo y predicción automática de futuros ciclos y días fértiles.
* Visualización de los datos en un calendario interactivo.
* Consulta de un historial con todos los registros anteriores.
* Gestión de cuenta de usuario con autenticación.
* Envío de recordatorios y notificaciones personalizadas.
* Acceso multiplataforma a través de la web.

Con estas funcionalidades, LifeFlow se presenta como una solución útil, accesible y adaptable para quienes deseen llevar un control personal de sus ciclos menstruales desde cualquier lugar y dispositivo.

1. **Diseño técnico**

El diseño técnico de LifeFlow, engloba todos los aspectos que organizan el desarrollo de la aplicación, desde la esquematización de la base de datos, hasta la propia interfaz de usuario.

La aplicación está construida de 2 formas: la parte front-end está dividida por las diferentes páginas de la aplicación, que a su vez compartes componentes ubicados en otro directorio.

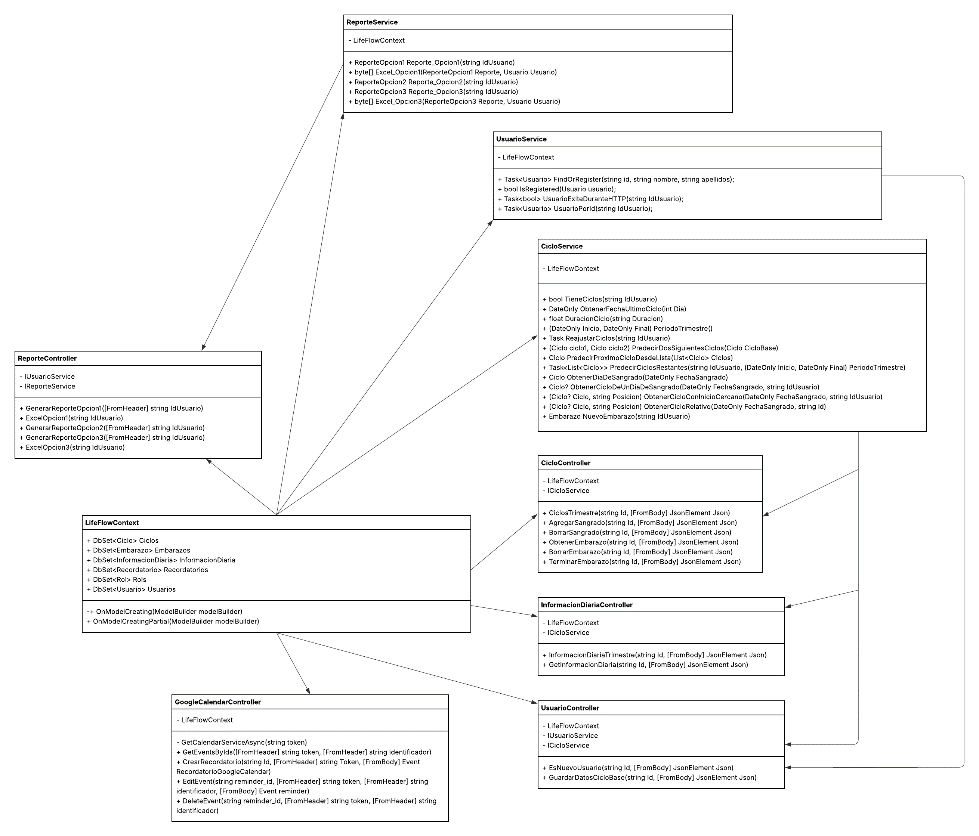
La parte back-end, mediante la API de .NET, está construido mediante una arquitectura MVC (Modelo-Vista-Controlador), un patrón de diseño que tiene como principio, que cada uno de los componentes esté separado en diferentes objetos, de forma que no se pueden combinar dentro de una misma clase: la vista representa la parte front-end, los modelos como el túnel que da entrada y salida a la base de datos, y el controlador, el responsable de gestionar y trabajar con los datos.

Si analizamos la solución ‘lifeflow-api’, la carpeta de modelos (Models > Scaffold), contienen todas las tablas de la base de datos en forma de objeto. Esto es gracias a la ingeniería inversa de Entity Framework: junto con una cadena de conexión y el paquete NuGet proovedor de la base de datos, Entity genera un contexto de base de datos y todas las tablas en forma de objetos para acceder a datos con objetos en vez de consultas. A esto se le llama ORM “Mapeo relacional de objetos”. De esos objetos, tienen asignado un controlador con todas las acciones disponibles, en forma de funciones que externamente, son acciones para consumir como API REST.

* 1. Diagrama de clases

El siguiente diagrama de clases, fue utilizado para el diseño técnico del desarrollo lógico del sitio web. Esta muestra las clases relacionadas con los modelos existentes en nuestro proyecto. Estas clases permiten recoger información de la base de datos, almacenarla y enviarla al controlador. El controlador, a su vez, administra la información recibida de los modelos y la distribuye a los lugares donde recibieron la petición.

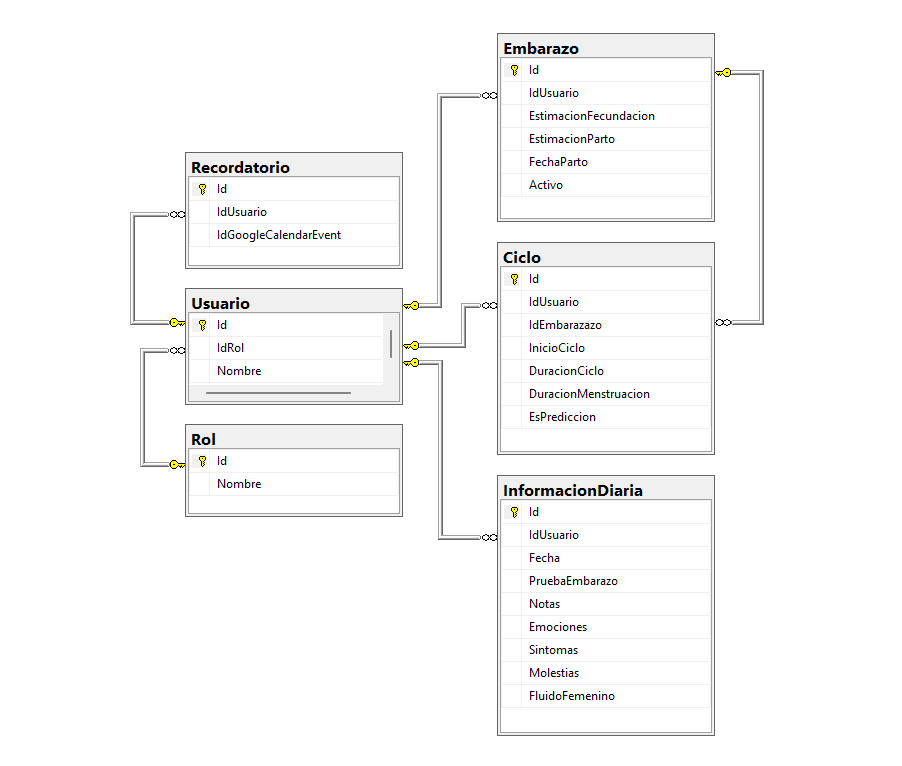
Para este desarrollo, se ha utilizado C# como lenguaje de programación del lado del servidor. C# es un lenguaje multiplataforma de uso general que hace que los desarrolladores sean productivos al escribir código de alto rendimiento, tiene una amplia compatibilidad con el ecosistema y todas las cargas de trabajo de .NET. Basado en principios orientados a objetos, incorpora muchas características de otros paradigmas, en particular de la programación funcional.

El esquema se centra en un contexto de base de datos, que funciona para trabajar en acceso a datos con objetos “ORM”. Está distribuido por todos los controladores y servicios. Los servicios actúan para funciones específicas de ese modelo para que los demás controladores puedan acceder a él, en vez de ser funciones exclusivas de cada controlador.

*Ilustración: Diagrama de clases (imagen con más resolución* [*aquí*](https://ibb.co/Xfmcfp28)*)*

* 1. Esquema de bases de datos

La creación de las tablas se ha basado en recolectar la información de los ciclos menstruales, almacenar los datos y su manipulación en el sitio, así como la gestión básica de usuarios. Estas tablas contienen la información indispensable, ya sea para mantener la privacidad o para asegurar una administración eficiente y segura del sitio web y la información personal y privada de los usuarios registrados para crear un sitio seguro y confiable.

El objetivo principal ha sido normalizar las tablas para reducir la redundancia de datos y facilitar su gestión.

*Ilustración: Esquema de base de datos*

* 1. Diseño de la interfaz del usuario

El diseño de la interfaz de LifeFlow, se define por una gama de colores rosas y azules, junto con un diseño simple y minimalista, con el objetivo de causar al usuario, sensación de confort visual y orden de los elementos que componen la misma.

La sencillez es la cualidad más importante de la aplicación, el objetivo era crear un diseño de interfaz web cómodo para cualquier usuario, sin sobrecargarla con detalles innecesarios y con la mayor accesibilidad posible.

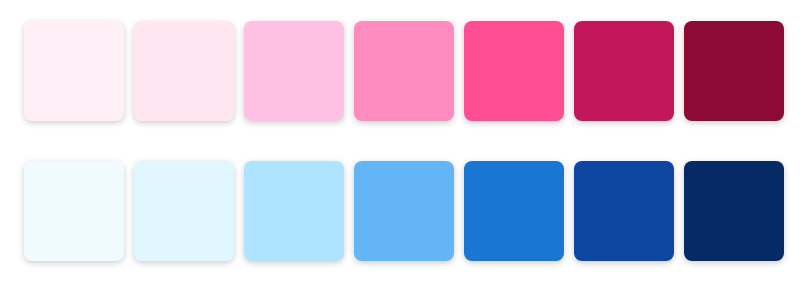
Para la aplicación de seguimiento de ciclos menstruales, se ha optado por una paleta de tonos rosados y azules cuidadosamente seleccionados.

Los tonos rosados evocan aspectos tradicionalmente asociados al ámbito femenino, al cuidado personal y al bienestar, transmitiendo suavidad, calidez y cercanía emocional. Además, los tonos más suaves generan una sensación de tranquilidad y comodidad en el uso diario.

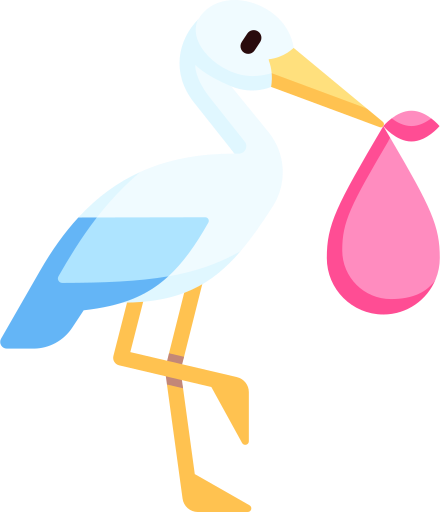
Los tonos azules complementan la paleta aportando estabilidad, confianza y profesionalismo. El azul es un color universalmente asociado a la salud y la tecnología, lo que ayuda a equilibrar el componente emocional de los tonos rosados con una percepción de rigor, fiabilidad y seriedad en el seguimiento de datos sensibles.

La combinación de ambos permite ofrecer una experiencia de usuario emocionalmente acogedora, visualmente armónica y profesionalmente confiable, muy adecuada para una aplicación de salud íntima como es el seguimiento del ciclo menstrual.

*Ilustración: Paleta de colores de LifeFlow*



Se eligió la cigüeña como logotipo por su fuerte asociación simbólica con la fertilidad, la maternidad y el ciclo de la vida. Este símbolo transmite de forma amable y accesible el propósito de la aplicación: acompañar a las usuarias en el seguimiento de su ciclo menstrual y salud reproductiva. La cigüeña aporta cercanía emocional, reduce la carga de un tema íntimo y genera confianza. Además, es un icono culturalmente reconocible y positivo, fácilmente recordable por las usuarias. De este modo, refuerza la identidad empática, optimista y respetuosa de la aplicación.



*Ilustración: Logotipo de LifeFlow*

1. **Tecnología utilizada**

En este apartado, se argumentarán todas las tecnologías utilizadas durante el trascurso del desarrollo de la aplicación junto con una justificación de la elección.

LifeFlow es un proyecto que se ha concebido como una oportunidad para aplicar tecnologías modernas y competitivas dentro del sector del desarrollo de software. Al mismo tiempo, ha servido como un espacio para potenciar la capacidad autodidacta y adquirir nuevas habilidades técnicas y profesionales relevantes para el futuro desarrollo laboral.

Para empezar, en las tecnologías implicadas en el desarrollo de la “parte visual”, destacan la librería React para trabajar con componentes web, SCSS para utilizar CSS en su máximo potencial, además de HTML5 para la construcción de la misma y JavaScript para realizar lógica en relación al front-end y consumición de la API para acceder a base de datos. La elección de las tecnologías aprendidas aplicadas, React y SCSS, ha sido porque se tratan de habilidades muy demandadas laboralmente, con una amplia comunidad de desarrolladores que aportan nuevas ideas, resuelven errores de otros y ayudan a que la misma comunidad aumente.

La parte visual fue previamente maquetada usando librerías de componentes e iconos para enfocar la interfaz a un estilo minimalista, limpio y agradable para cualquier vista. Esas librerías son: Material Icons, FlatIcon, Bootstrap, Notyf. También se ha aprovechado la oportunidad para desarrollar mis habilidades gráficas para crear imágenes conforme a la identidad de la aplicación: minimalismo y paleta azul-rosa.

La lógica de LifeFlow es una parte fundamental del sistema, ya que gestiona el procesamiento de los datos, las predicciones del ciclo menstrual y la generación de reportes. Para su desarrollo se utilizan dos tecnologías principales: JavaScript en el frontend y C# con .NET en el backend. JavaScript es el lenguaje principal del lado del cliente. Se utiliza para crear una experiencia de usuario dinámica, gestionar la interacción con la interfaz, controlar el comportamiento del navegador y realizar peticiones a la API. Permite aplicar parte de la lógica de negocio directamente en el frontend y facilita la comunicación con el servidor. .NET es un framework desarrollado por Microsoft, utilizado en este proyecto para construir la API y gestionar la lógica del lado del servidor. En conjunto con el lenguaje C#, permite organizar el sistema usando el patrón MVC. Se encarga de procesar la información, generar predicciones y exponer los servicios que el frontend consume.

LifeFlow utiliza servicios de Google Cloud Platform para implementar funcionalidades clave relacionadas con la autenticación y la integración con herramientas externas. En concreto, se ha habilitado el inicio de sesión mediante cuentas de Google, lo que facilita un acceso seguro y rápido para los usuarios sin necesidad de crear una cuenta específica en la aplicación. Esta autenticación se gestiona a través del cliente de Google OAuth configurado en Google Cloud. Además, la aplicación integra Google Calendar para que los usuarios puedan sincronizar y visualizar fechas importantes relacionadas con su ciclo menstrual directamente en su calendario personal. Esta integración mejora la experiencia al centralizar la información y facilitar recordatorios automáticos.

Para mejorar la experiencia del usuario, la aplicación tiene implementado la funcionalidad de informar al usuario de los eventos que van surgiendo durante su uso, como la creación exitosa de recordatorios o el registro de días de sangrado. Esa funcionalidad se lleva a cabo gracias a la librería Notyf, que ayuda a mostrar información al usuario en forma de alertas minimalistas y cómodas a la vista.

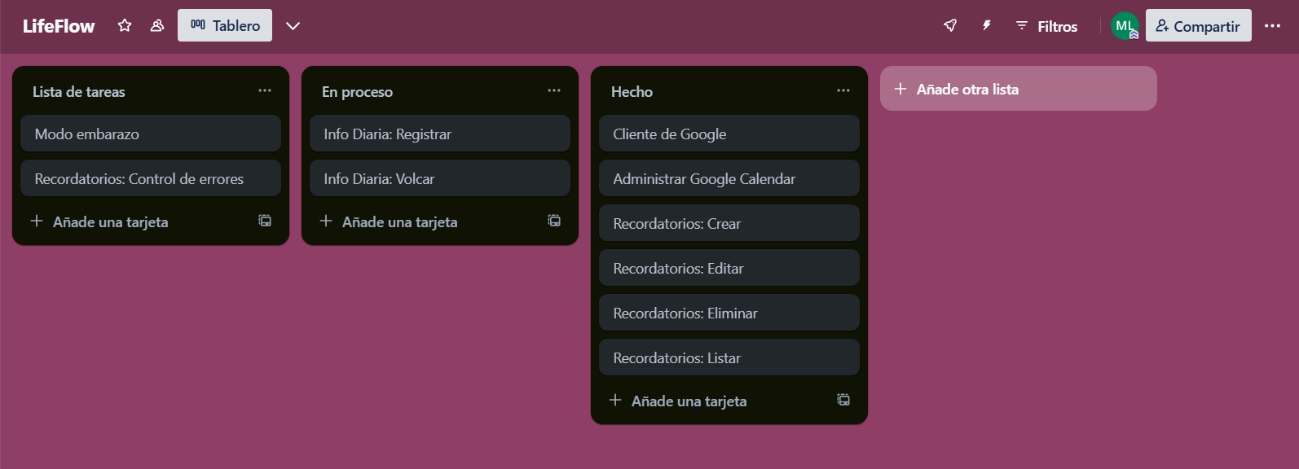
SQL Server es el proveedor de la base de datos de la aplicación, se trata de un gestor de bases de datos relacionales desarrollada por Microsoft. Es la base de datos más compatible con la API de .NET ya que ambos fueron desarrollados por la misma compañía, por lo que subirla a un entorno de producción o en la nube, sería más fácil de integrar que con otras bases de datos relacionales.

Para terminar, el usuario tiene la posibilidad de exportar sus reportes en forma de Excel, gracias a la dependencia ClosedXML dentro del catálogo de paquetes NuGet que ofrece el IDE Visual Studio a la hora de desarrollar en .NET.

1. **Planificación**

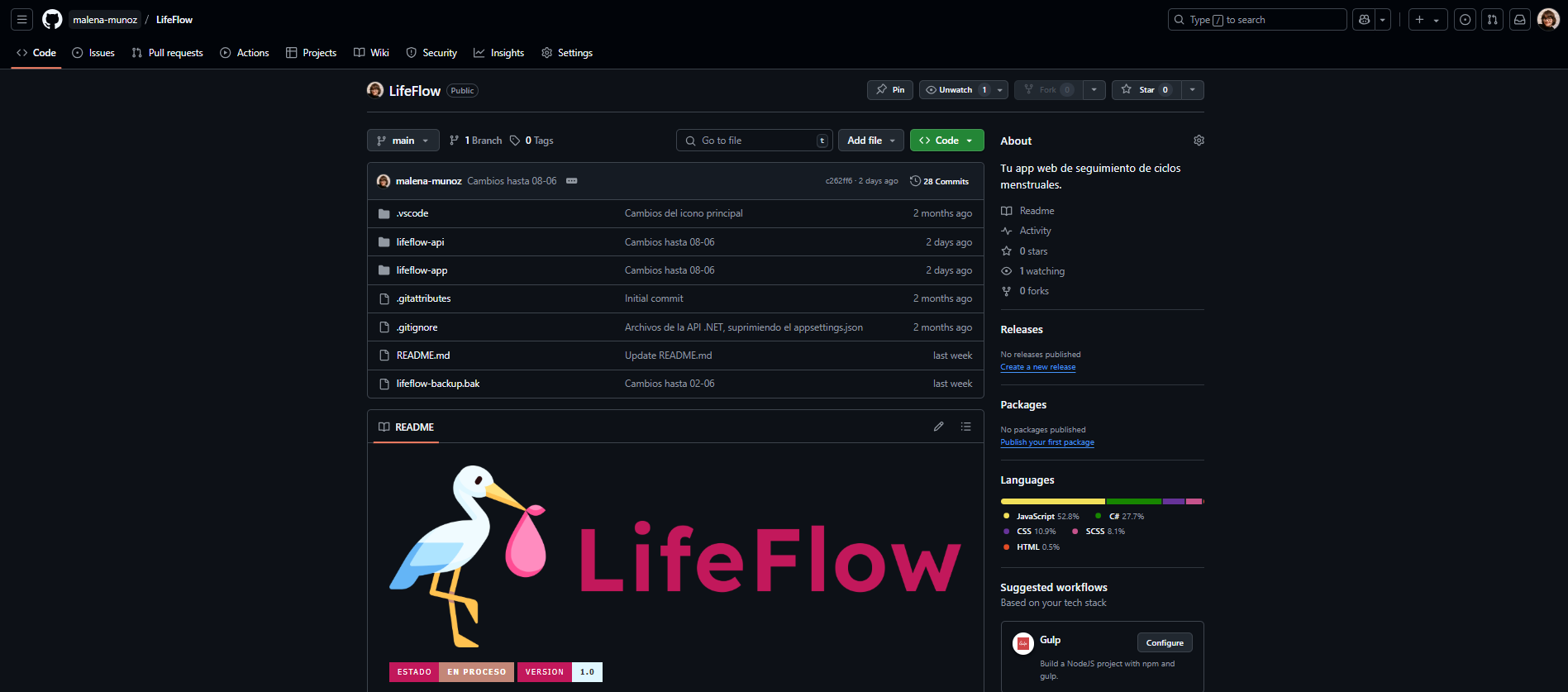
La planificación de tareas se fijó durante el seguimiento de las funcionalidades que tendría la aplicación, las cuales no se establecieron ninguna duración, y conforme se iban terminando, otras se empezaban. Al no ser un proyecto grande que involucra muchos recursos, lo mejor era seguir la metodiogía Kanban para trabajar con flujos de trabajo.

Para mantener guardado el flujo del trabajo, se ha utilizado Trello para crear y administrar las tareas que definían nuevos requerimientos o funcionalidades para la aplicación.



*Ilustración: Tablero de Trello con las tareas que fueron programadas a mitad de mayo*

A su vez, el control del código se ha guardado y administrado con git, y Github como la interfaz para tener la aplicación subida a un repositorio en la nube y así, poder acceder a él desde cualquier dispositivo. Para acceder haz clic [aquí](https://github.com/malena-munoz/LifeFlow).



*Ilustración: Repositorio del proyecto en Github*

Las tareas se han dividido principalmente por las vistas que tiene la aplicación, y a su vez, por los módulos que la forma.

1. Página de bienvenida
   1. Diseño de interfaz
   2. Acceso al inicio de sesión con Google
2. Página principal
   1. Diseño de interfaz
   2. Registro de información diaria
   3. Volcado de información diaria previa
   4. Control de embarazo
   5. Interacción de la información
   6. Control de los ciclos: verificar si terminaron, registro y eliminación de sangrados, cálculo de las predicciones menstruales
3. Recordatorios
   1. Diseño de la interfaz
   2. Configurar los servicios de Google Calendar
   3. Registrar recordatorios
   4. Editar recordatorios
   5. Visualizar recordatorios
   6. Borrar recordatorios
4. Reportes
   1. Diseñar reportes acordes a la información registrada
   2. Recogida lógica de la información
   3. Diseño de interfaz
   4. Investigación acerca del ciclo menstrual
5. Guía
   1. Diseño de interfaz
6. Privacidad
   1. Diseño de interfaz
7. **Presupuesto**

El siguiente presupuesto está orientado a un posible caso real de cuánto costaría hacer LifeFlow si un cliente desea solicitar la aplicación. Los precios estarán basados en todos los aspectos a tener en cuenta durante su hipotético desarrollo, como los costes del cliente, el salario del trabajador, espacio de trabajo, alquiler y hosting web.

Tecnologías: Todas las tecnologías usadas, son gratuitas, por lo que son de libre uso para la comunidad y no supondrían ningún coste para el cliente. En otro caso, lo que puede suponer un costo sería el cliente de Google, encargado del inicio de sesión y controlar eventos mediante Google Calendar.

El cliente de Google que viene de Google Cloud, puede cobrarte servicios adicionales si se supera el límite de invocaciones establecidos del plan gratuito. Lo ideal es poner 5€ mensuales para que Google te avise sobre cuando ese límite va a terminar y evitar cargos no deseados, como un seguro.

Hosting web: Teniendo en cuenta que el front-end está desarrollado con React y el back-end con .NET, las opciones son muy variadas, como Azure, Railway, etc. En este caso, la mejor opción sería subir el front-end en Vercel y la API de .NET en Render, junto con la base de datos de SQL Server. El precio estimado rondaría sobre los 12€ al mes.

Personal: Basándonos en mi salario actual, un perfil junior con un año de experiencia laboral y un contrato de 40 horas semanales, el salario bruto anual sería de 2000€.

Alquiler de equipo y lugar de desarrollo: El equipo constaría de un portátil alquilado por cualquier empresa de equipos informáticos, rondan normalmente por 30€ al día. Teniendo en cuenta un mes de 30 días, el coste mensual sería de 900€. En cuanto al lugar de trabajo, el espacio estaría ubicado el Alcorcón ya que es el lugar de domicilio de la programadora. Constaría de un alquiler mensual de 110€ aproximadamente.

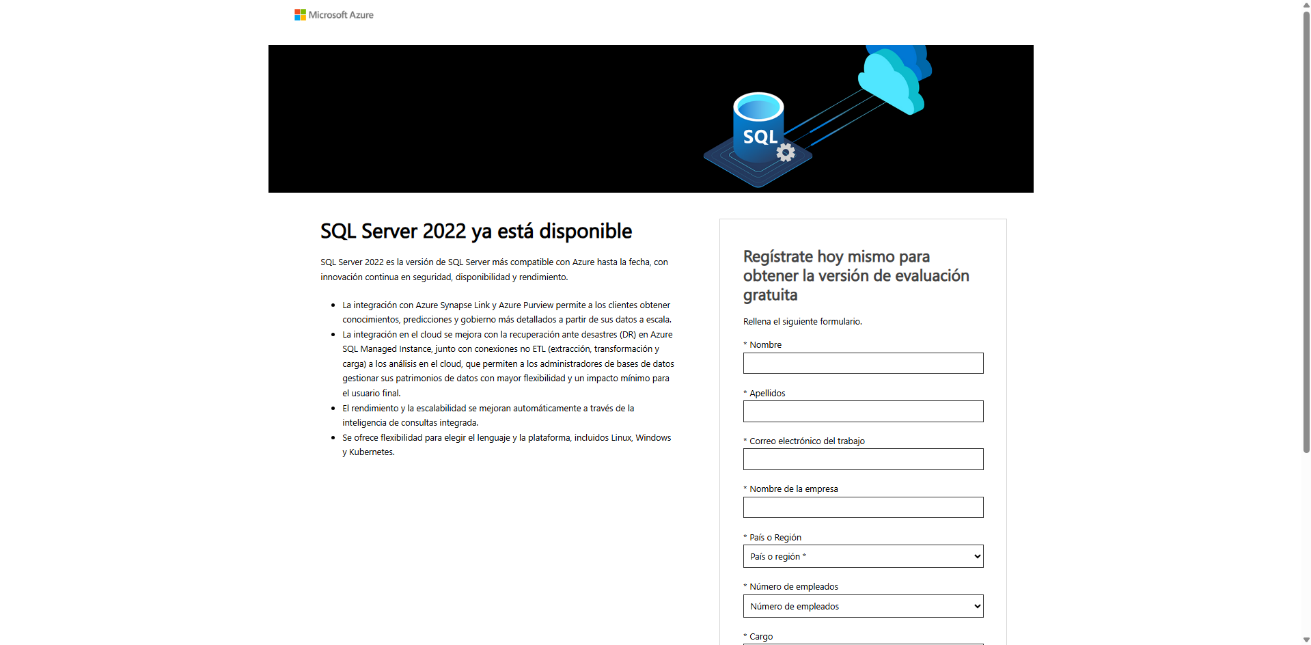
Una vez se calcule el presupuesto mensual, es importante tener en cuenta que el hosting web y el seguro del cliente de Google son costes fijos que son necesarios para el funcionamiento de la aplicación.

|  |  |
| --- | --- |
| **Descripción** | **Coste mensual (€)** |
| Seguro del cliente de Google | 5€ |
| Hosting web | 12€ |
| Salario de la programadora | 2000€ |
| Alquiler de portátil | 900€ |
| Alquiler de oficina de trabajo | 110€ |
| Total | 3027 € |

El coste por el desarrollo de LifeFlow oscilan los 3027€, recalcando los costes mensuales fijos necesarios para mantener la aplicación operativa.

1. **Manual de instalación**

El manual de instalación incluye toda la información necesaria para que el usuario pueda configurar la aplicación desde un ordenador, puesto que actualmente no se encuentra en un entorno de producción.

Para empezar, el primer paso sería instalar la base de datos SQL. Lo primero es acceder a <https://info.microsoft.com/ww-landing-sql-server-2022.html?lcid=es-es> y rellenar el formulario:

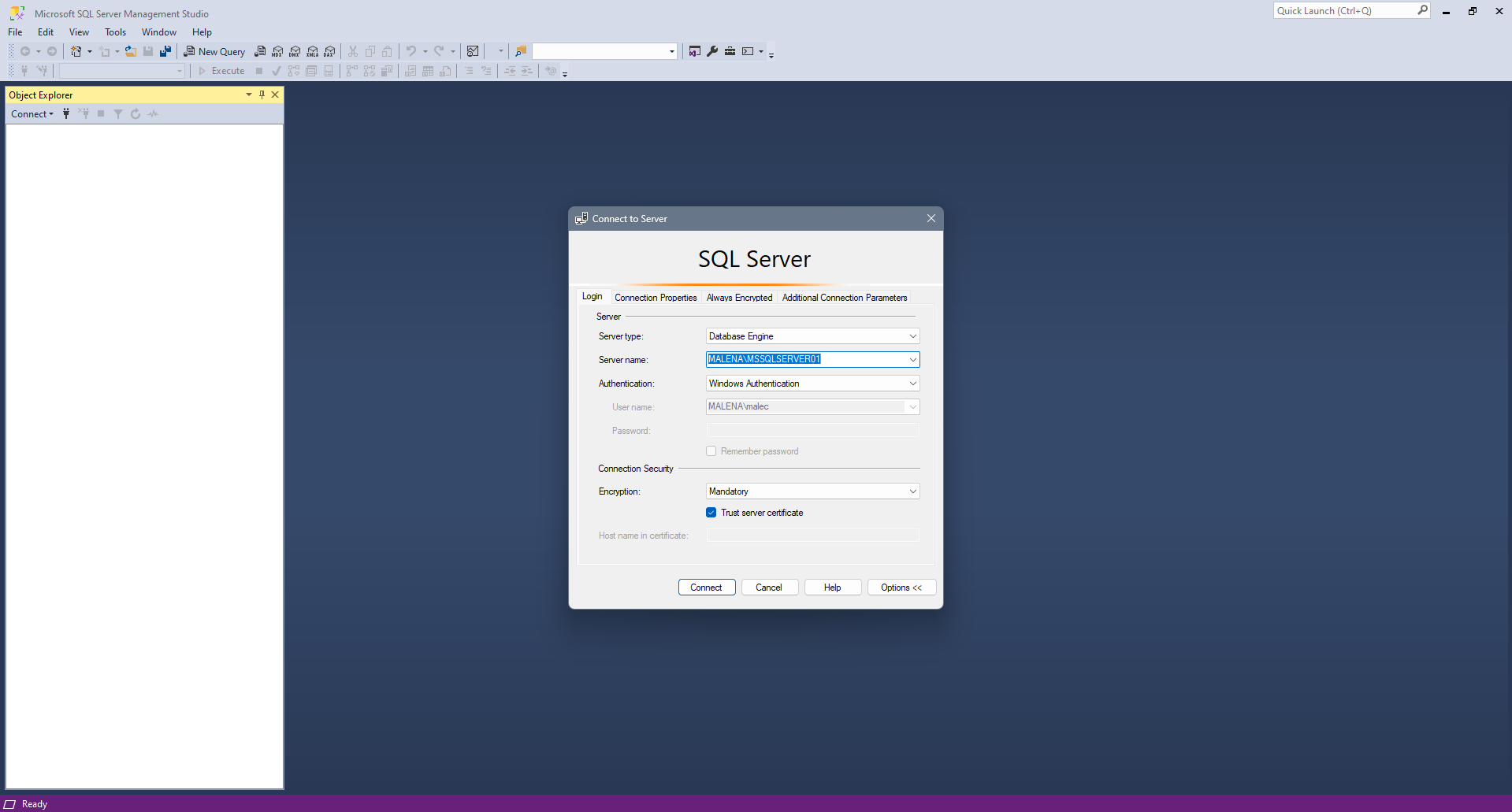
*Ilustración: Formulario previo a la descarga de SQL Server*

Una vez rellenado, sigue al botón de descargar ahora. Aparecerá una página para seleccionar que versión quieres descargarte. Elegir la descarga por EXE.

Al descargar el archivo y al abrirlo, te ofrecerá 3 tipos diferentes de instalación: básica, personalizada y descargar medios. Elegir la descarga básica. Una vez escogido el tipo de descarga, aceptamos la declaración de privacidad y seleccionamos el directorio donde se quiere descargar (lo mejor es dejarlo por defecto). Es posible que tengas que reiniciar el ordenador durante la instalación.

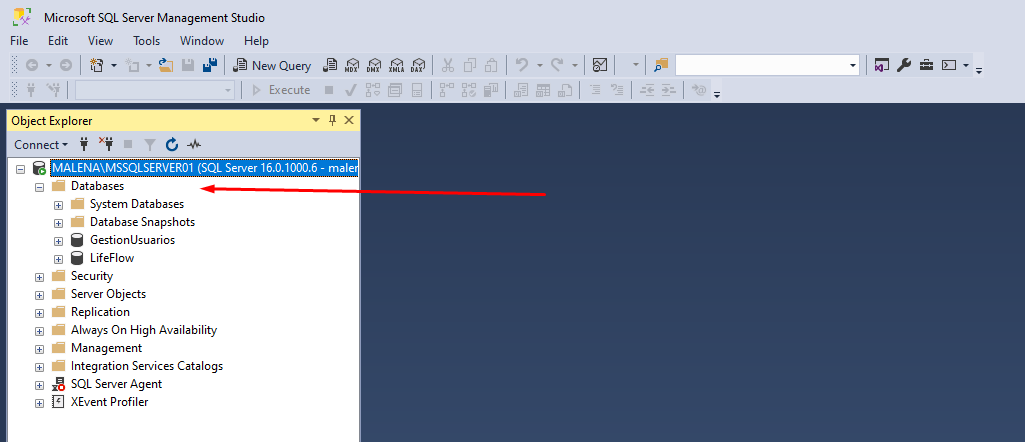
Cuando SQL Server esté instalado, el programa te preguntará que, si quieres instalar SSMS, que es la interfaz para trabajar con SQL Server y T-SQL. Acéptalo y espera a que se descargue.

Entra en SSMS si tienes tanto SQL Server como SSMS instalado. Aparecerá una vista similar a la siguiente:

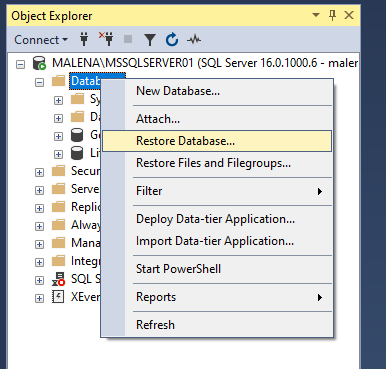


*Ilustración: Inicio al entrar a SSMS*

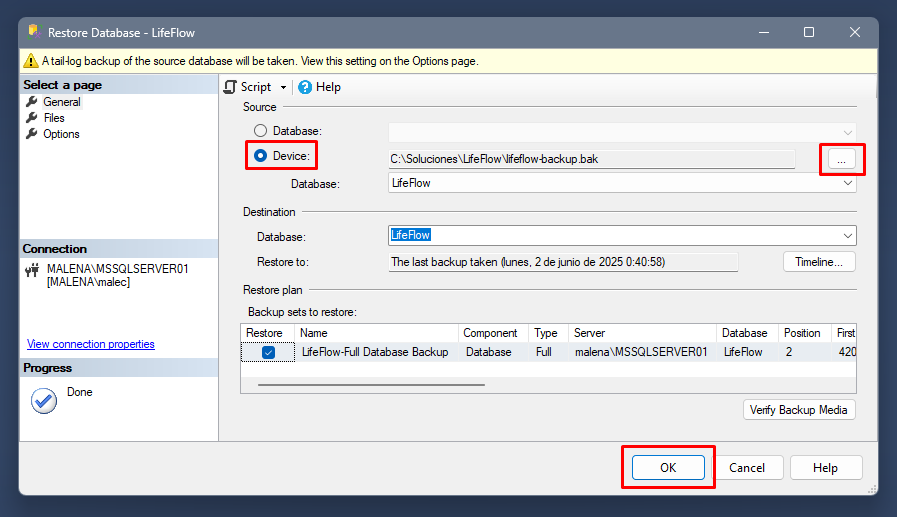
Si no lo tenías previamente instalado, entra mediante la autenticación de Windows y acepta el certificado del servidor, ya que no te pide ninguna contraseña. Cuando accedes, estás accediendo a un servidor, que en este caso sería tu ordenador. Puedes acceder a diferentes módulos, pero el que nos interesa es el de “Databases”, que es donde se alojará nuestra base de datos.



*Ilustración: Módulo de bases de datos dentro del servidor en SSMS*

Haciendo clic derecho en la carpeta de “Databases”, hay una opción llamada “Restore Database…”. En esa opción deberemos sunir nuestro archivo ‘lifeflow-backup.bak’ para importar la base de datos de LifeFlow.

*Ilustración: Opción “Restore Database…” para importar la base de datos de la aplicación*

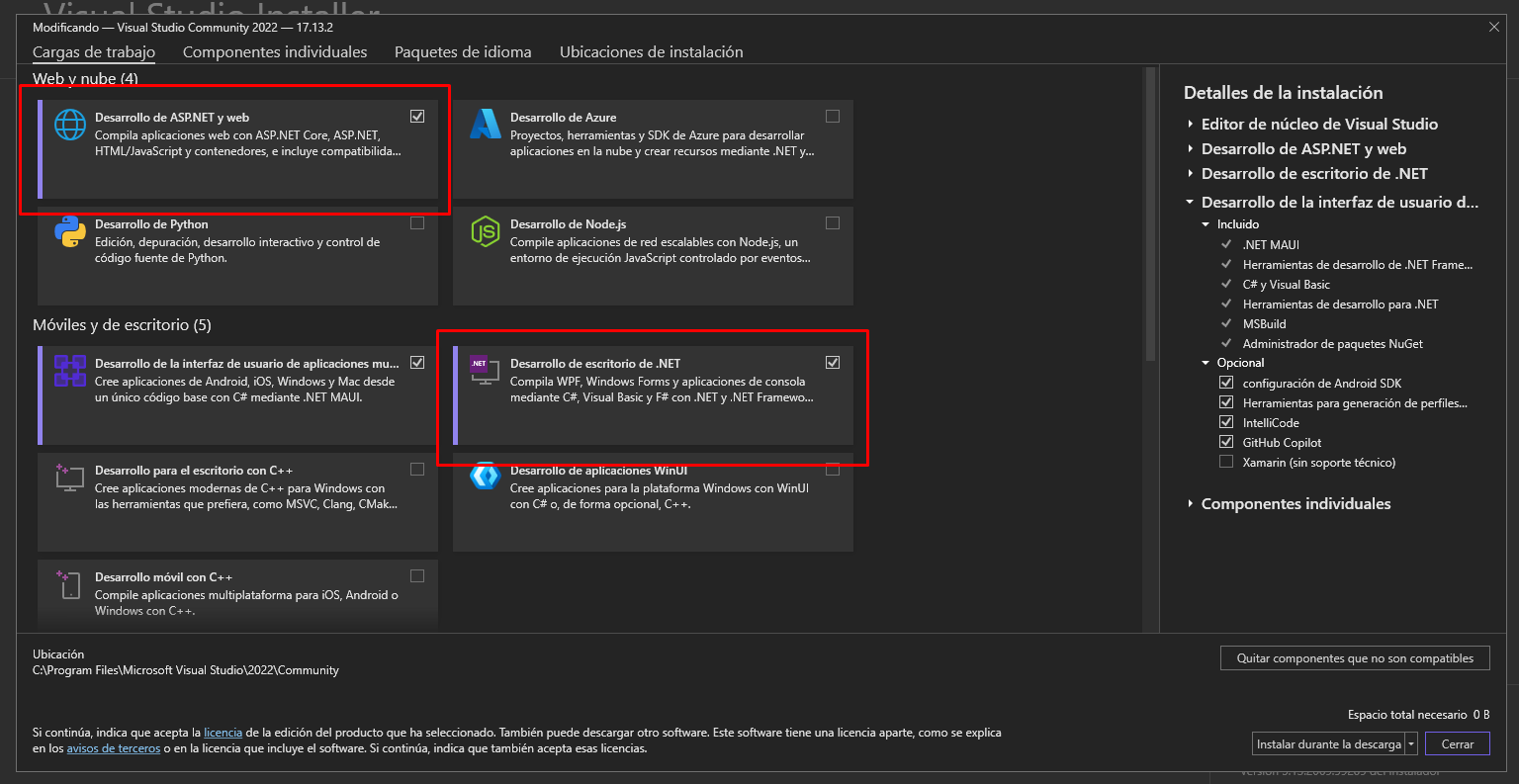
1. Seleccione ‘Device’ y luego, el botón de puntos suspensivos (…) para buscar el archivo ‘lifeflow-backup.bak’. Agrega el archivo una vez lo hayas encontrado.
2. Seleccione OK para restaurar la copia de seguridad de la base de datos. La base de datos ya estaría importada.

*Ilustración: Importar el backup de la base de datos de LifeFlow en SSMS*

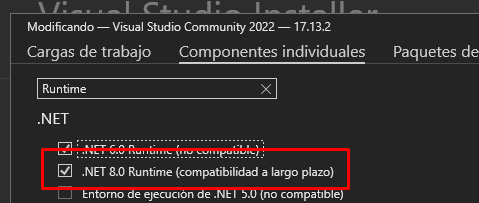
Para instalar la API, necesitaremos Visual Studio 2022 y posteriormente, instalar las opciones necesarias para el desarrollo en .NET.

Microsoft ofrece 3 diferentes descargas de Visual Studio en <https://visualstudio.microsoft.com/es/downloads/>, pero a nosotros nos interesa la versión de la comunidad. Ejecutamos el EXE cuando esté descargado y vamos aceptando la licencia y la declaración de privacidad conforme avancemos. Después de instalar el Instalador de Visual Studio, puede usarlo para personalizar la instalación seleccionando los conjuntos de características o cargas de trabajo, que desee.

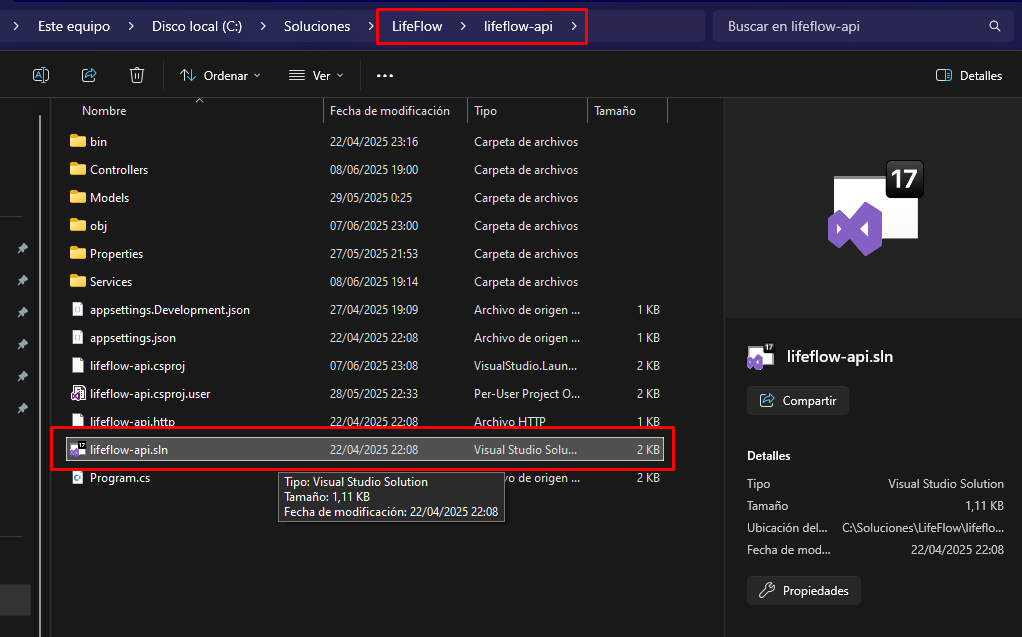
A nosotros nos interesa las cargas de Desarrollo de ASP.NET y web, y Desarrollo de escritorio de .NET, junto con la versión .NET 8.0 Runtime ubicado en componentes individuales.



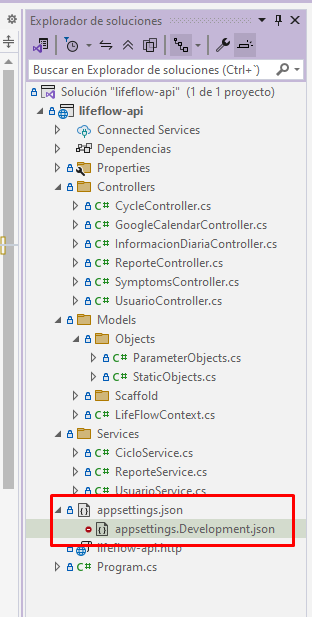
*Ilustración: Cargas de trabajo para la instalación*



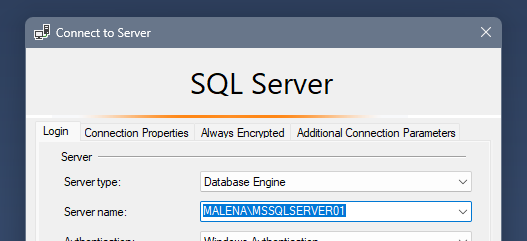
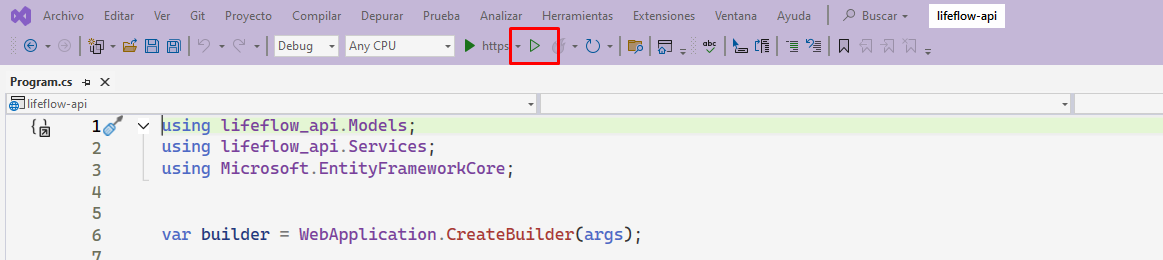
*Ilustración: Versión de .NET para la instalación*

Después de seleccionar los requisitos, le hacemos clic al botón inferior derecho para instalar todo. Ya tendrías el IDE y .NET listo para lanzar la API. Tan solo sería abrir el archivo solución dentro de la carpeta ‘lifeflow-api’ de la aplicación, cambiar el servidor de la cadena de conexión a SQL Server al del dispositivo actual y ejecutarlo sin depuración.

*Ilustración: Ubicación del archivo solución para abrir la API en Visual Studio*

Para cambiar el servidor de la cadena de conexión a SQL Server, tienes que ir a los ajustes de la aplicación en desarrollo ‘appsettings.Development.json’ y cambiar el parámetro ‘Data Source’ al de tu actual dispositivo (equivale a ‘Server name’ cuando vas a conectar al servidor en SSMS).

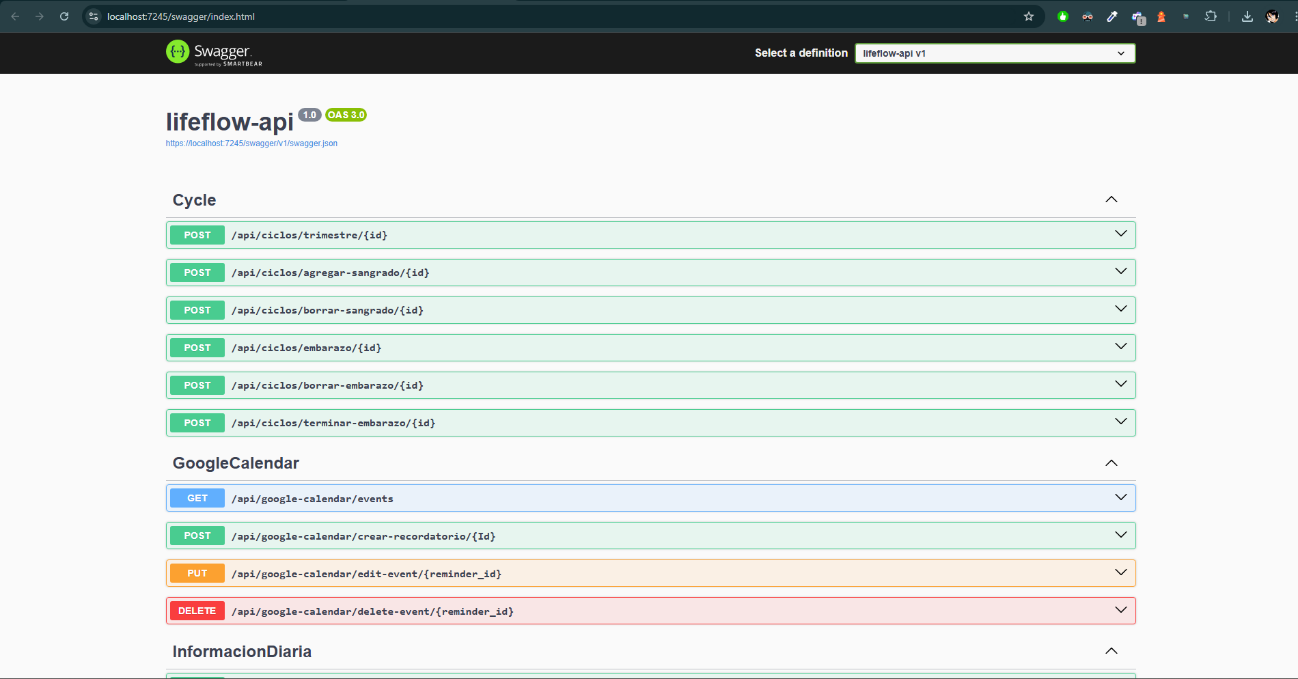
*Ilustración: Ubicación de ‘appsettings.Development.json’ en el explorador de soluciones*



*Ilustración: Ejecutar el modo sin depuración para iniciar la API*

*Ilustración: Cambiar el parámetro ‘Data Source’*

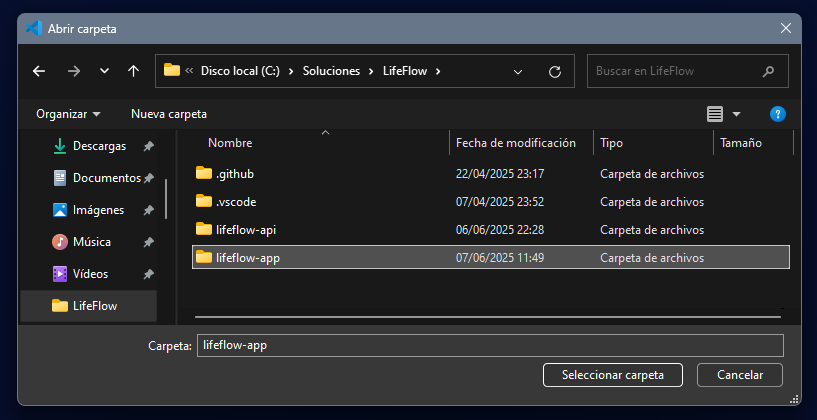
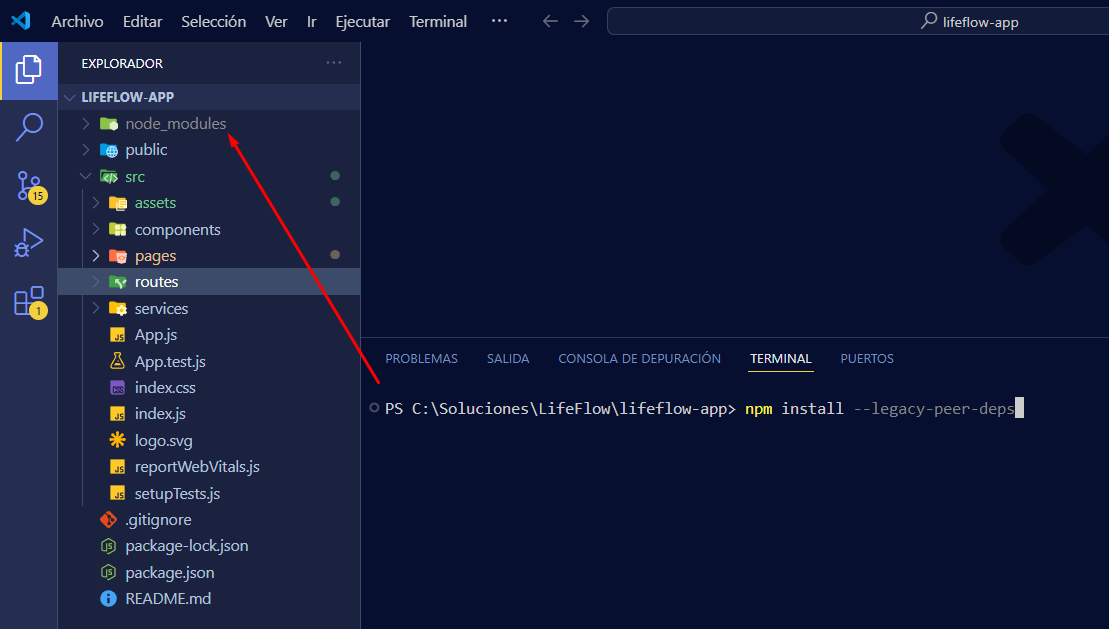
Al ejecutarlo, Visual Studio compila la solución para revisar si no tiene errores previos. La API estaría operativa cuando aparezca la pestaña de Swagger (sirve para probar las peticiones).



*Ilustración: Pestaña de Swagger al ejecutar la API*

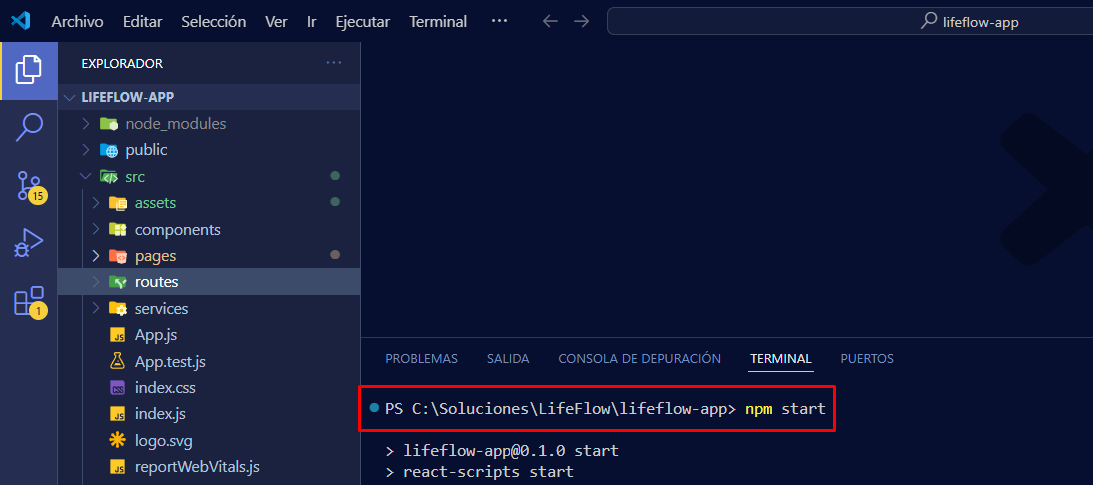
Por último, sería descargar las dependencias del proyecto y lanzarlo. Para ello, usaremos Visual Studio Code, que se puede descargar en <https://code.visualstudio.com/download>.

1. Instalamos la versión de Windows y una vez descargada, abrimos el IDE.
2. Nos dirigimos en el menú superior izquierdo, donde aparece las opciones Archivo, Editar, Selección, etc. Nos vamos a Archivo > Abrir carpeta…
3. Seleccionamos la otra parte de la aplicación ‘lifeflow-app’, que contiene la aplicación web en React.
4. Cuando la app esté abierta, abrimos una terminal y ejecutamos el comando ‘npm install –legacy-peer-deps’, que instala todas las dependencias. **Es necesario tener Node (vía** [**https://nodejs.org/es/download**](https://nodejs.org/es/download)**)**
5. Con las dependencias instaladas, ejecutar en la terminal ‘npm start’ para lanzar la aplicación.



*Ilustración: Ejecutar ‘npm install –legacy-peer-deps’. Instalaría la carpeta ‘node\_modules’*

*Ilustración: Abrimos la carpeta ‘lifeflow-app’*



*Ilustración: Ejecutar ‘npm start’*

1. **Bibliografía**

Esta son algunas de las páginas en las que he podido recaudar información, documentación y conocimientos que han sido de ayuda para el desarrollo de la aplicación, y la investigación de los ciclos menstruales y el embarazo:

<https://flo.health/es/tu-ciclo>

<https://medlineplus.gov/spanish/>

<https://learn.microsoft.com/es-es/ef/core/managing-schemas/scaffolding/?tabs=dotnet-core-cli>

<https://react-bootstrap.netlify.app/>

<https://www.flaticon.es/>

<https://es.react.dev/>

<https://carlosroso.com/notyf/>

<https://datatables.net/>

<https://stackoverflow.com/questions>