**Proyecto CalGrow**

1. ***Objetivo del proyecto***

Crear un sistema automatizado el cual permita la recolecta de cultivos de forma más sencilla, eficiente y segura, siendo capaz de registrar datos sobre el clima, humedad del suelo, generar informes y mantener a los técnicos informados para las alertas críticas mediante el uso de Tecnologías de la Información.

***Requisitos Funcionales***

**R.F-01:** El sistema debe conectarse a los sensores IoT presentes en la zona.

**R.F-02:** El sistema debe integrarse con estaciones meteorológicas públicas.

**R.F-03:** El sistema debe realizar una lectura de los datos enviados por los sensores IoT cada 15 minutos.

**R.F-04:** El sistema debe generar un informe en formato Excel/PDF sobre los datos entregados por los sensores IoT y enviarlos a las auditorías.

**R.F-05:** El sistema debe conectarse con los drones presentes en la zona.

**R.F-06:** El sistema debe informar alertas tempranas de plagas mediante las imágenes que provean los drones.

**R.F-07:** El sistema debe enviar notificaciones push para alertas críticas.

**R.F-08:** El sistema debe tener un mapa interactivo con las zonas de riesgo actuales.

**R.F-09:** El sistema debe tener un modo offline.

**R.F-10:** El sistema debe hacer gráficos comparativos.

***Requisitos No Funcionales***

**R.N.F-01:** Interfaz rápida, intuitiva y sencilla de entender.

**R.N.F-02:** Tiempo de respuesta menor a 2 segundos.

**R.N.F-03:** Escalable a más de 100 hectáreas.

**R.N.F-04:** Cumplir con la protección para datos de proveedores.

**R.N.F-05:** Tener una disponibilidad del 96% mensual.

**R.N.F-06:** Procesar gran variedad de datos sin bajones de rendimiento.

**R.N.F-07:** Permitir incorporación de nuevas parcelas o tipos de cultivos sin afectar la infraestructura del sistema.

**R.N.F-08:** Código fuente documentado y estructurado para mantenimiento y expansión del sistema.

**R.N.F-09:** Proteger acceso y permisos según roles.

**R.N.F-10:** En caso de falla del sistema, garantizar recuperación en menos de 5 segundos.

***Presupuesto Global de Proyecto***

El proyecto, CalGrow, debido a las complejas funcionalidades y la gran cantidad de integrantes en el equipo tendrá un presupuesto de $21.431.823 de pesos, más el 10% es de $23.813.137 de pesos.

***Roles en el Equipo***

Desarrollador FullStack: Profesional capaz de trabajar tanto en el frontend (interfaz de usuario) como en el backend (lógica del servidor y base de datos) de una aplicación.

Sueldo Promedio: $1.400.000 – $2.400.000

Desarrollador Backend: Especialista en la lógica del servidor, bases de datos, APIs y arquitectura que permiten que una aplicación funcione detrás de escena.

Sueldo Promedio: $1.600.000 – $1.800.000

Desarrollador Frontend: Encargado de construir la parte visual e interactiva de una aplicación web, usando tecnologías como HTML, CSS y JavaScript.

Sueldo Promedio: $1.200.000 – $1.600.000

Jefe de Proyecto: Responsable de la planificación, ejecución y cierre del proyecto. Supervisa equipos, cronogramas, presupuesto y calidad.

Sueldo Promedio: $2.000.000 – $3.000.000

Gestor de Proyectos: Similar al jefe de proyecto, pero más enfocado en la coordinación general y la gestión administrativa y estratégica del proyecto.

Sueldo Promedio: $1.800.00 - $2.500.000

|  |
| --- |
|  |

Diseñador UI: Diseña la interfaz gráfica de una aplicación, asegurando que sea visualmente atractiva, coherente y fácil de usar.

Sueldo Promedio: $1.000.000 – $1.500.000

Tester: Se encarga de verificar que el software funcione correctamente mediante pruebas funcionales, de rendimiento y usabilidad, detectando errores antes del lanzamiento.

Sueldo Promedio: $1.000.000 - $1.500.000

Analista de Negocios: Actúa como puente entre el negocio y el equipo técnico, recolectando requisitos, entendiendo necesidades y asegurando que el software aporte valor.

Sueldo Promedio: $1.500.000 – $2.200.000

Product Owner: Define y prioriza los requisitos del producto (backlog), asegurando que el equipo desarrolle lo que tiene mayor valor para el negocio y los usuarios.

Sueldo Promedio: $1.800.000 – $2.800.000

Ingeniero DevOps: Automatiza, integra y mantiene los procesos de desarrollo, pruebas y despliegue para lograr una entrega continua y eficiente.

Sueldo Promedio: $1.800.000 – $2.500.000

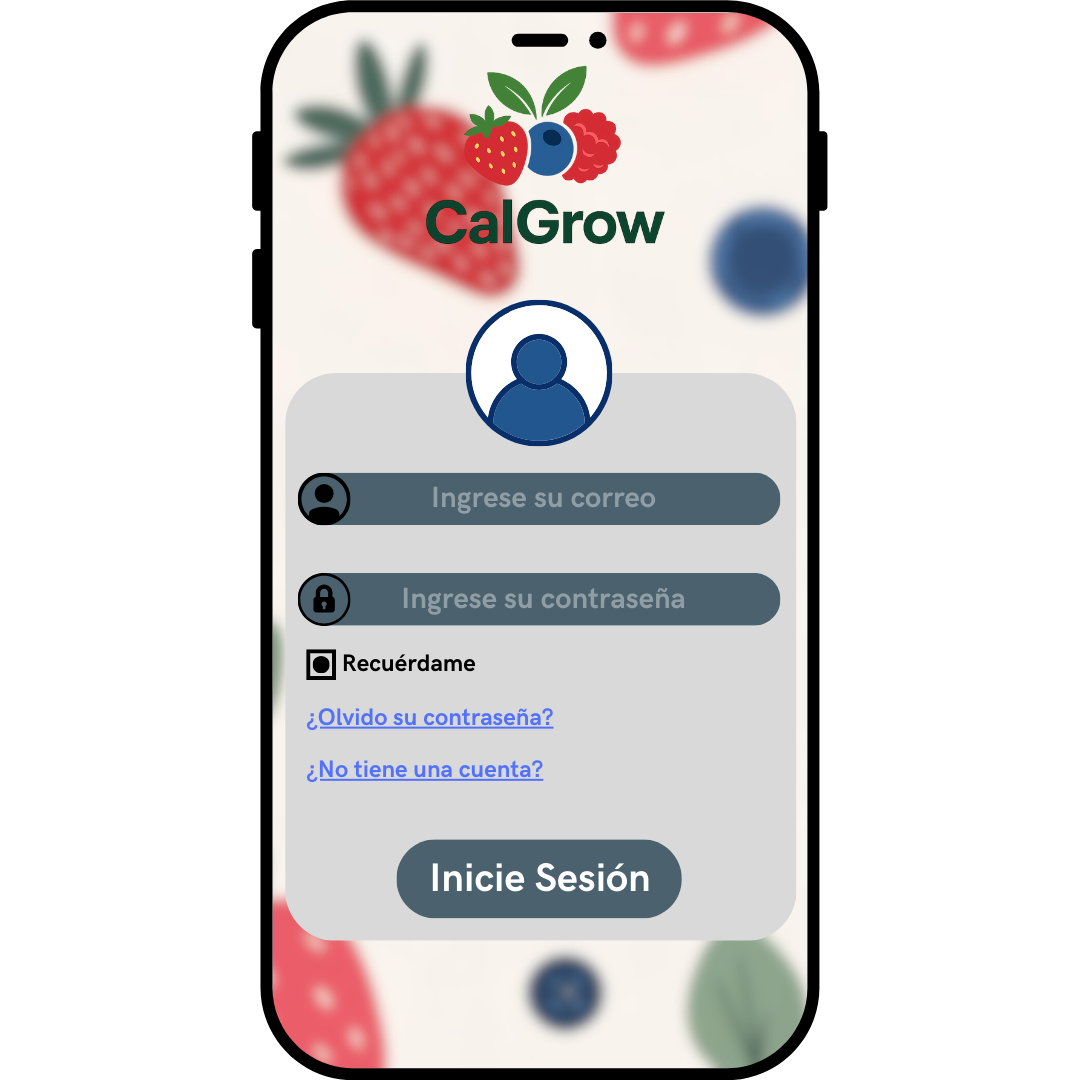
Desarrollador Mobile: Desarrolla aplicaciones para dispositivos móviles (iOS, Android), utilizando herramientas como Swift, Kotlin o frameworks multiplataforma como Flutter o React Native.

Sueldo Promedio: $1.400.000 - $2.000.000

***Metodología Utilizada***

Nuestro equipo utiliza la metodología en Cascada, es decir, se va construyendo el proyecto de forma secuencial, primero terminando una etapa para continuar con las siguientes y así.

***Mockup***



|  |
| --- |
|  |

|  |
| --- |
|  |

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

|  |
| --- |
|  |

|  |
| --- |
|  |

***Diagramas UML***

A continuación, se mostrará un diagrama UML que representa la lógica y funcionamiento de nuestro sistema por cada vista.

***Diagrama de Clases***

Un diagrama de clases es una representación visual del diseño de un sistema orientado a objetos, donde se muestran las clases, sus atributos, métodos y las relaciones entre ellas (como herencia, asociación o dependencia). Este Diagrama corresponde a la vista lógica del sistema:

Diagrama

El contenido generado por IA puede ser incorrecto.

***Diagrama de Componentes***

Un diagrama de componentes es una representación visual que muestra cómo se organizan y conectan los módulos físicos o lógicos de un sistema, llamados *componentes* (por ejemplo, bibliotecas, servicios o ejecutables), y sus dependencias. Este Diagrama corresponde a la vista de despliegue del sistema:

Diagrama

El contenido generado por IA puede ser incorrecto.

***Diagrama MER***

Un diagrama MER es una representación gráfica que muestra las entidades relevantes de un sistema, sus atributos y las relaciones que existen entre ellas, con el objetivo de diseñar y estructurar una base de datos de forma lógica. Este Diagrama corresponde a la vista física del sistema:

Diagrama, Texto

El contenido generado por IA puede ser incorrecto.

***Diagrama de Caso de Uso***

Un diagrama de casos de uso es una representación visual que muestra las funciones o acciones que un sistema ofrece (los *casos de uso*) y los actores que interactúan con él, destacando qué funcionalidades están disponibles para cada tipo de usuario. Este Diagrama corresponde a la vista de escenarios del sistema:

Diagrama

El contenido generado por IA puede ser incorrecto.

***Diagrama de Procesos***

Un diagrama de procesos es una representación gráfica que describe la secuencia de actividades o tareas que componen un proceso, mostrando cómo se transforman insumos en resultados, y cómo fluye la información o el trabajo entre pasos. Este diagrama corresponde a la vista de procesos del sistema:

Imagen que contiene Gráfico

El contenido generado por IA puede ser incorrecto.Diagrama

El contenido generado por IA puede ser incorrecto.

***Justificación del diseño según ISO/IEC 25010***

El diseño del sistema CalGrow se basa en la norma ISO/IEC 25010 para asegurar la calidad del software. Se priorizan características como funcionalidad (cumplimiento de requisitos como integración con sensores IoT y generación de alertas), usabilidad (interfaz intuitiva), eficiencia (tiempos de respuesta inferiores a 2 segundos), fiabilidad (disponibilidad del 96%) y mantenibilidad (código documentado y estructurado). Estas cualidades garantizan que el sistema sea robusto, escalable y alineado con los objetivos del proyecto.

***Conclusiones***

El proyecto CalGrow dará como resultado una mejor gestión para los cultivos de berries de la empresa AgroFértil S.A, no solo logrando una mayor eficiencia y disminución de los problemas que presentan actualmente, sino que, también, logrando la automatización de la mayoría de procesos que ejecuta la empresa.