



Evidencia. Documento de la solución del reto

Aplicación 2: Horas de calor

Integración de seguridad informática en redes y sistemas de software
(Gpo 570)

Equipo 4

Daira Adriana Chavarría Rodríguez

Kevin Ramiro Duque Molina

Nicolás Herrera Hernández

María Teresa Angulo Tello

Carolina Valdés Guerra

Prof. Alejandro García Ruiz

Prof. Roberto A. Guevara González

6 de mayo de 2022

Instituto Tecnológico y de Estudios Superiores de Monterrey

Control de Versiones

Fecha	Versión	Descripción del cambio	Autor(es)	Autorizó
29/Abr/22	v1.00	Es la primera iteración del documento	Equipo 4	Todo el equipo
05/May/22	v2.00	Completación del documento	Equipo 4	Todo el equipo

Contenido

Etapas 1. Requerimientos	5
Introducción	5
Necesidades del negocio	5
Propósito del producto	5
Alcance del producto	5
Descripción del producto	5
Perspectiva del producto	5
Funcionalidades del producto	5
Características de los usuarios	5
Limitaciones, supuestos y /o dependencias	5
Requerimientos específicos	6
Requerimientos Funcionales	6
Casos de Uso	7
Requerimientos de datos lógicos	9
Restricciones de diseño	9
Glosario	10
Referencias	10
Anexos	11
Aportación Individual y en equipo	11
Gantt/Cronograma del reto.	12
Matriz de riesgos	13
Etapas 2. Diseño	14
Diagrama del diseño de la Base de datos.	14
Diagrama de la arquitectura de la app.	15
Diagrama de clases de la app.	16
Diseño de las interfaces de usuario.	17
Integración de seguridad en sistemas computacionales.	18
Etapas 3. Desarrollo	19
URL del repositorio de software con el código del servidor, código del cliente.	19
Configuración de la infraestructura necesaria para desarrollar el proyecto.	19
Documentación de reuniones con el cliente, desarrolladores y equipo de pruebas.	19
Identificación de ataques informáticos.	20
Métodos de protección de ataques informáticos.	20
Etapas 4. Pruebas	22
Documento de estrategia de pruebas.	22
Plan de pruebas.	24

Casos de prueba y datos de prueba	27
RTM (Requirement Traceability Matrix)	28
Reporte con resumen y resultados de las pruebas.	28
Informe de cierre de las pruebas.	29
Reporte de incidentes.	29
Etapas 5. Despliegue	33
Planeación de la migración del software.	33
Manual de usuario.	33
Manual de instalación	36
Capacitación.	37
5. Publicación de la app.	37

Etapa 1. Requerimientos

1. Introducción

1.1. Necesidades del negocio

Actualmente DiGiSKY requiere una aplicación que permita mostrar en un dispositivo iOS la información relacionada con la cantidad de horas calor que lleva una planta agrícola. Las unidades de calor son la cantidad de horas en las cuales se registró una temperatura superior a 10 grados Celsius. Considerando que el negocio está involucrado en el área agrícola tener estos datos monitoreados es de gran importancia.

1.2. Propósito del producto

Se busca crear una aplicación para facilitar la información para el dueño de la parcela y así dar el seguimiento adecuado a la parcela deseada.

El enfoque de la solución se realiza a partir del modelo de la base de datos actual, simplificando varios aspectos para el Usuario (Dueño de la parcela), donde ya no es necesario la presencialidad del equipo de DiGiSky para que se le dé el seguimiento adecuado de las horas calor a las parcelas contempladas.

1.3. Alcance del producto

El modelo interactivo estará presentado en un aplicación en el que los usuarios de AgroSky pueden encontrar información sobre sus parcelas en cuanto a horas calor respecta, estos contenidos están orientados a la gente con una o varias parcelas. El producto permitirá ser ejecutado en dispositivos IOS y recordará el historial de las horas calor entre las distintas parcelas vinculadas.

2. Descripción del producto

2.1. Perspectiva del producto

El producto será una aplicación para dispositivos de la compañía Apple con el sistema operativo iOS, la aplicación será diseñada para ser accesible desde cualquier lugar siempre y cuando este cuente con internet para realizar la descarga de información pertinente.

2.2. Funcionalidades del producto

Este sistema permitirá mostrar las últimas actualizaciones de la base de datos respecto a las horas calor que la parcela designada reciba, permitiendo que se mejore el seguimiento para el desarrollo adecuado de las plantaciones agrícolas, de la misma manera que se le puede dar el seguimiento sin importar la ubicación que se encuentre el Usuario.

2.3. Características de los usuarios

Los usuarios objetivos son las personas que se encuentran registradas dentro de la compañía de AgroSky, al mismo tiempo que se le permite el acceso a personas terceras designadas por el Usuario (Dueño de la parcela), para que le de un seguimiento adecuado a las parcelas deseadas, en este caso como Usuarios encargado.

2.4. Limitaciones, supuestos y /o dependencias

La aplicación estará diseñada para dispositivos específicos con iOS. Aparte de esto, el producto estará diseñado asumiendo que el Usuario (Dueño de la parcela y encargado), conoce lo que afecta las horas calor de la parcela prevista, siendo capacitados por los miembros de DiGiSky.

3. Requerimientos específicos

3.1. Requerimientos Funcionales

01. Inicio de sesión	
ID Req.	Requerimiento Funcional
RF01001	La aplicación usará un sistema de autenticación con correo y contraseña para el acceso a la misma, se contará con un pantalla de inicio de sesión donde se aplicará el proceso antes descrito.
RF01002	El usuario será asignado por el administrador una contraseña con el correo proporcionado al ser cliente de DiGiSKY, con estos datos se realizará el inicio de sesión.
RF01003	El usuario podrá recuperar su contraseña mediante el correo electrónico asignado a la cuenta.
Comentario	<p>Para el caso de los requerimientos RF01001 y RF01003 quedan fuera de nuestro alcance por falta de tiempo, siendo así que queda a manera de propuesta en futuras implementaciones si en dado caso así se desea.</p> <p>Para el caso del requerimiento RF01002 quedará fuera de nuestro alcance como desarrolladores.</p>

02. Visualización de horas calor	
ID Req.	Requerimiento Funcional
RF02001	El usuario podrá visualizar las horas calor de la parcela seleccionada. La visualización de las horas calor será mediante una tabla donde se muestran las horas calor acumuladas en el día.
RF02002	El usuario podrá cambiar entre las distintas parcelas permitidas en el momento que lo desee.
RF02003	El sistema validará que la información esté en la última actualización de la base de datos, siempre y cuando exista una conexión a internet, en caso contrario desplegará la última información cargada.

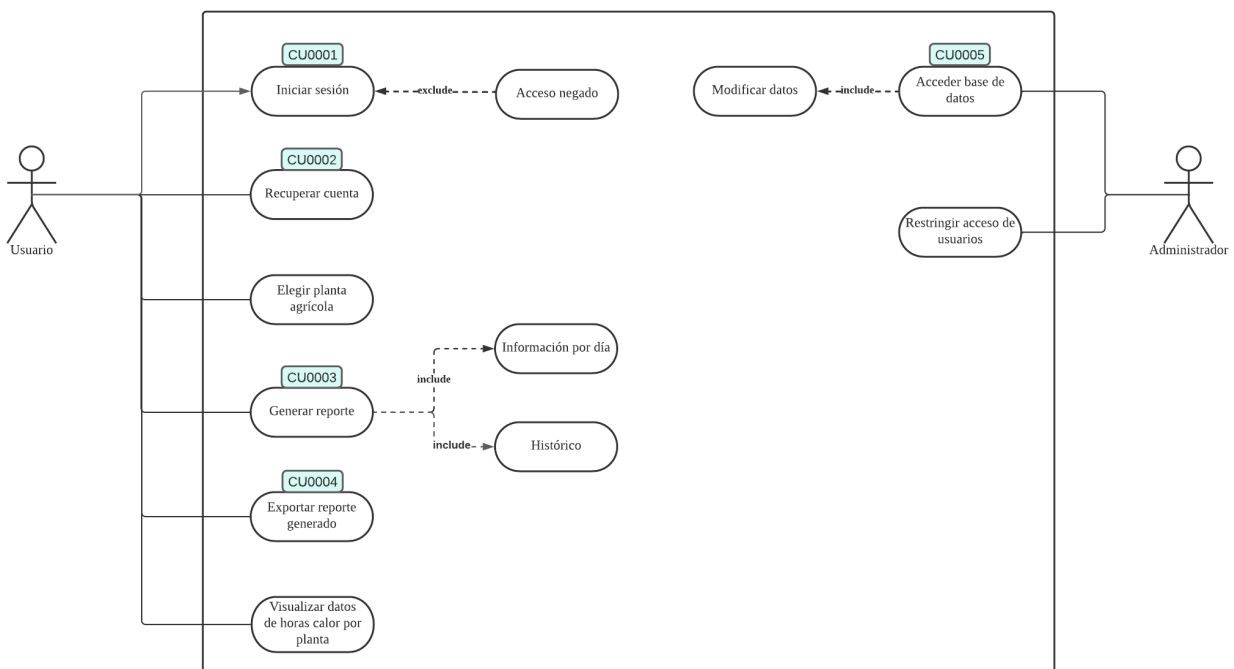
03. Generación de reportes	
ID Req.	Requerimiento Funcional
RF03001	La aplicación generará un reporte detallando las horas calor de una parcela.

RF03002	El usuario podrá cambiar el periodo de tiempo para la representación en el reporte, cuando así lo desee.
RF03003	El sistema validará que la información sea apropiada para ser exportada.

04. Exportación de reportes.	
ID Req.	Requerimiento Funcional
RF04001	Posterior a la generación del reporte, este se podrá exportar como un documento con formato PDF para su uso fuera de la aplicación.
RF04002	El usuario podrá decidir a donde quiere exportar el documento PDF, ya sea por correo electrónico, guardar al dispositivo, mensaje, etc.
RF04003	El usuario podrá cancelar la exportación del documento y regresar a la pantalla anterior.

3.2. Casos de Uso

3.2.1. Diagrama UML



CU0001. Inicio de sesión	
Actor principal	Usuario
Precondición	El usuario debe contar con una cuenta existente dentro de nuestra base de datos y deberá encontrarse dentro de la aplicación

Escenario Principal	<ol style="list-style-type: none"> 1. El usuario ingresa su correo electrónico. 2. El usuario ingresa la contraseña 3. El usuario realizará un toque en el botón de “Inicio de sesión” y es llevado a la pantalla home de la aplicación
Escenarios de excepción	En dado caso que el correo electrónico o la contraseña sean incorrectos la aplicación deberá mostrar una alerta notificando el error y solicitando que se llenen los campos de información nuevamente.
Comentarios	El proceso de “Inicio de Sesión” queda fuera de nuestro alcance debido al corto periodo de tiempo. Sin embargo, está propuesto para un desarrollo recomendado si se le desea dar seguimiento a la aplicación.

CU0002. Recuperar cuenta	
Actor principal	Usuario
Precondición	El usuario debe contar con una cuenta existente dentro de nuestra base de datos y deberá encontrarse dentro de la aplicación
Escenario Principal	<ol style="list-style-type: none"> 1. El usuario realizará un toque en “¿Olvidó su contraseña?”. 2. El sistema muestra la pantalla de recuperar cuenta. 3. El usuario ingresa el correo electrónico deseado a recuperar. 4. El usuario dará un toque en recuperar cuenta. 5. El sistema deberá enviar un correo para la recuperación.
Escenarios de excepción	En dado caso que el correo electrónico sea inexistente el sistema deberá desplegar una alerta informando el error y pidiendo al usuario que vuelva a ingresar los datos necesarios.
Comentarios	El proceso de “Recuperar cuenta” queda fuera de nuestro alcance debido al corto periodo de tiempo, sin embargo esta propuesto para un desarrollo recomendado si se le desea dar seguimiento a la aplicación.

CU0003. Generación de reporte	
Actor principal	Usuario
Precondición	CU0001
Escenario Principal	<ol style="list-style-type: none"> 1. Seleccionar parcela de la cuál se quiera generar reporte 2. Usuario elige entre opción del reporte del día o histórico 3. Aplicación calcula las horas calor basándose a partir de una fórmula predeterminada 4. Se despliega en la información en la pantalla.
Escenarios de excepción	Sin escenario de excepción
Comentario	Para el caso de la precondición CU0001 que está fuera del alcance se le indicará a la aplicación un usuario previo para realizar las pruebas pertinentes.

CU0004. Exportación de reporte	
Actor principal	Usuario
Precondición	Cumplir con CU0003
Escenario Principal	<ol style="list-style-type: none"> 1. En ventana donde se generó el reporte en el CU0002, seleccionar exportar reporte 2. Aplicación genera PDF y muestra las opciones disponibles para la exportación 3. Usuario elige la ubicación deseada a guardar o enviar
Escenarios de excepción	<ol style="list-style-type: none"> 1. Falta de espacio en dispositivo: <ol style="list-style-type: none"> a. Despliega mensaje de error por falta de espacio

CU0005. Acceso a base de datos	
Actor principal	Administrador
Precondición	Tener permiso de administrador / cuenta de empleado de DiGiSky.
Escenario Principal	<ol style="list-style-type: none"> 1. El administrador ingresará a Firebase desde el sitio Web. 2. Agregar la información que sea pertinente.
Escenarios de excepción	Escenarios sin excepciones
Comentarios	Este caso de uso está fuera de nuestro alcance como desarrolladores.

3.3. Requerimientos de datos lógicos

01. Recuperar cuenta	
ID Req.	Requerimiento de datos lógicos
RD01001	Toda contraseña de usuario debe tener al menos una mayúscula, un número y un carácter especial.
RD01003	Se guardará la contraseña vinculada al correo electrónico proporcionado siendo actualizado en la base de datos de Firebase.

3.4. Restricciones de diseño

01. Software y lenguajes a utilizar	
ID Req.	Restricción de diseño
RR01001	La aplicación será desarrollada para funcionar en IOS.
RR01002	La vinculación de la Base de Datos deberá ser con Firebase siendo proveída por el socio formador.

02. Funcionalidad

ID Req.	Restricción de diseño
RR02001	La aplicación está diseñada para ser usada desde dispositivos IOS, ya sea iPhone o iPad.
RR02002	La aplicación no requerirá de mucha capacidad para ser ejecutada en su totalidad.
RR02003	La aplicación sólo representará la información provista por la base de datos sin que ésta sea alterada.

3.5. Requerimientos no funcionales:

01. Operatividad del sistema	
ID Req.	Requerimiento no funcional
RN01001	La aplicación debe de ser programada en el lenguaje Swift
RN01002	La base de datos almacena la información en Firebase.
RN01003	Solo el personal autorizado por DiGiSKY tendrá acceso a la base de datos directamente.

02. Seguridad	
ID Req.	Requerimiento no funcional
RN02001	La aplicación debe de contar conexiones seguras a través de SSL
RN02002	La aplicación generará una copia de la base datos para evitar la alteración de los mismos.
RN02003	Toda la información dentro de la aplicación deberá ser encriptada con las herramientas que provee Swift para evitar que aplicaciones terceras dentro del dispositivo ingresen a la información.

4. Glosario

Término	Explicación
BD	Base de datos
Usuario	Persona que está haciendo uso de la aplicación
Administrador	Persona con una cuenta de empleado dentro de DiGiSky con permisos de administrador.
Home	La pantalla principal de nuestra aplicación.

5. Referencias

- Este documento se realizó siguiendo los estándares en requerimiento de software de la IEEE

- [El presente documento fue elaborado siguiendo las pautas establecidas en SWEBOK v.3 y estándar de especificación de requerimientos de la fábrica de software del tecnológico de monterrey v.1]

6. Anexos

6.1. Aportación Individual y en equipo

- 6.1.1. **Daira Adriana Chavarría Rodríguez:** Hubo un buen intercambio de opiniones con respecto a lo esperado de la aplicación, empezando por la decisión de qué proyecto desarrollar. Asimismo, hubo una buena comunicación tanto dentro como fuera de las sesiones de clase para conocer el estado de nuestro trabajo (no solamente de esta misma etapa). Individualmente, se trabajó principalmente en la corrección del documento y en la construcción de requerimientos.
- 6.1.2. **Kevin Ramiro Duque Molina:** El proceso para la creación del documento fue a partir del intercambio de ideas entre el equipo, siendo bajo un proceso iterativo de preguntas y respuestas tanto al profesor, al equipo, inclusive al socio formador, esto para lograr una correcta estructuración de los documentos, delimitando nuestros alcances en el corto periodo de tiempo disponible para la creación de la app y lograr de esa manera nuestras metas deseadas. En mi caso en particular desarrollé la parte de los requerimientos no funcionales, datos lógicos, diseños y el diagrama de Gantt; sin embargo cabe aclarar que todo el proceso fue con ayuda de mi equipo.
- 6.1.3. **Nicolás Herrera Hernández:** De acuerdo a la problemática establecida por el cliente y las necesidades resultantes de ella, se pudieron crear los requerimientos. Primero se establecieron los límites de nuestra aplicación, es decir, que íbamos a incluir y que no, ya que contamos con un tiempo muy corto para el desarrollo y al mismo tiempo se busco la manera de incluir todo lo que el cliente pedía. Trabajé de manera individual en la definición de los requerimientos funcionales y la matriz de riesgo, para definir los requerimientos tuvimos que ponernos de acuerdo como equipo.
- 6.1.4. **María Teresa Angulo Tello:** Habiendo sido informados acerca de los requerimientos para la aplicación del cliente, discutimos para organizar el desarrollo del proyecto. Con ello, delegamos las tareas para lo que se debía llevar a cabo, definimos una cronología para las actividades, y enlistamos dudas para tener todo preparado para antes de comenzar a trabajar. Mi aporte dentro de la entrega fue apoyar con el desarrollo de los diagramas de casos de uso, y verifiqué que el resto del documento estuviera correcto, comunicándome constantemente con mi equipo.

- 6.1.5. **Carolina Valdés Guerra:** Para elaborar este documento nos comunicamos sobre cuál era nuestra visión para este proyecto basándonos en lo que habíamos escuchado del socio formador y las preguntas que le hicimos para aclarar la idea que teníamos sobre lo que querían y cómo resolverlo, ya con eso nos pudimos concentrar en desarrollar este documento por partes. Yo trabajé principalmente en los casos de uso, pero para poder elaborarlos tuve que comunicarme constantemente con el resto del equipo.
- 6.1.6. **Aportación en equipo:** Como equipo utilizamos el tiempo en clase para primero aclarar lo que entendimos sobre el proyecto y lo propuesto por el socio formador y decir la visión que teníamos cada uno para poder resolverlo. Después de tener esta información y analizar la base de datos que se nos proporcionó pudimos ponernos de acuerdo sobre cómo sería la mejor forma de desplegar esta información en la aplicación. A lo largo de este proceso todos los integrantes participaron en tanto ideas como aportaciones en el proyecto.

6.2. Gantt/Cronograma del reto.

	Semana 1							Semana 2						
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Recopilar información	■	■												
Requerimientos		■	■	■	■	■								
Casos de usos		■	■	■	■	■								
Implementación						■	■	■	■	■	■			
Testing						■	■	■	■	■	■			
Pruebas de seguridad											■	■	■	
Preparación para presentación													■	■
Presentación														■

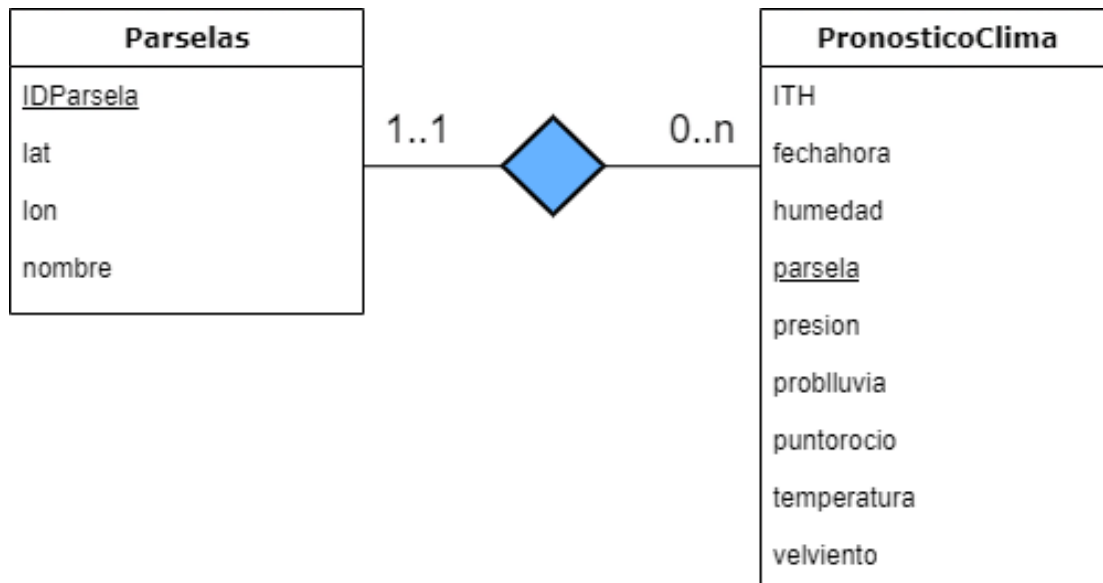
6.3. Matriz de riesgos

Valor	Probabilidad				Nivel de riesgo	
1	Raro				Muy alto	Rojo
2	Improbable				Alto	Naranja
3	Posible				Medio	Amarillo
4	Probable				Bajo	Verde
5	Casi seguro				Muy bajo	Azul

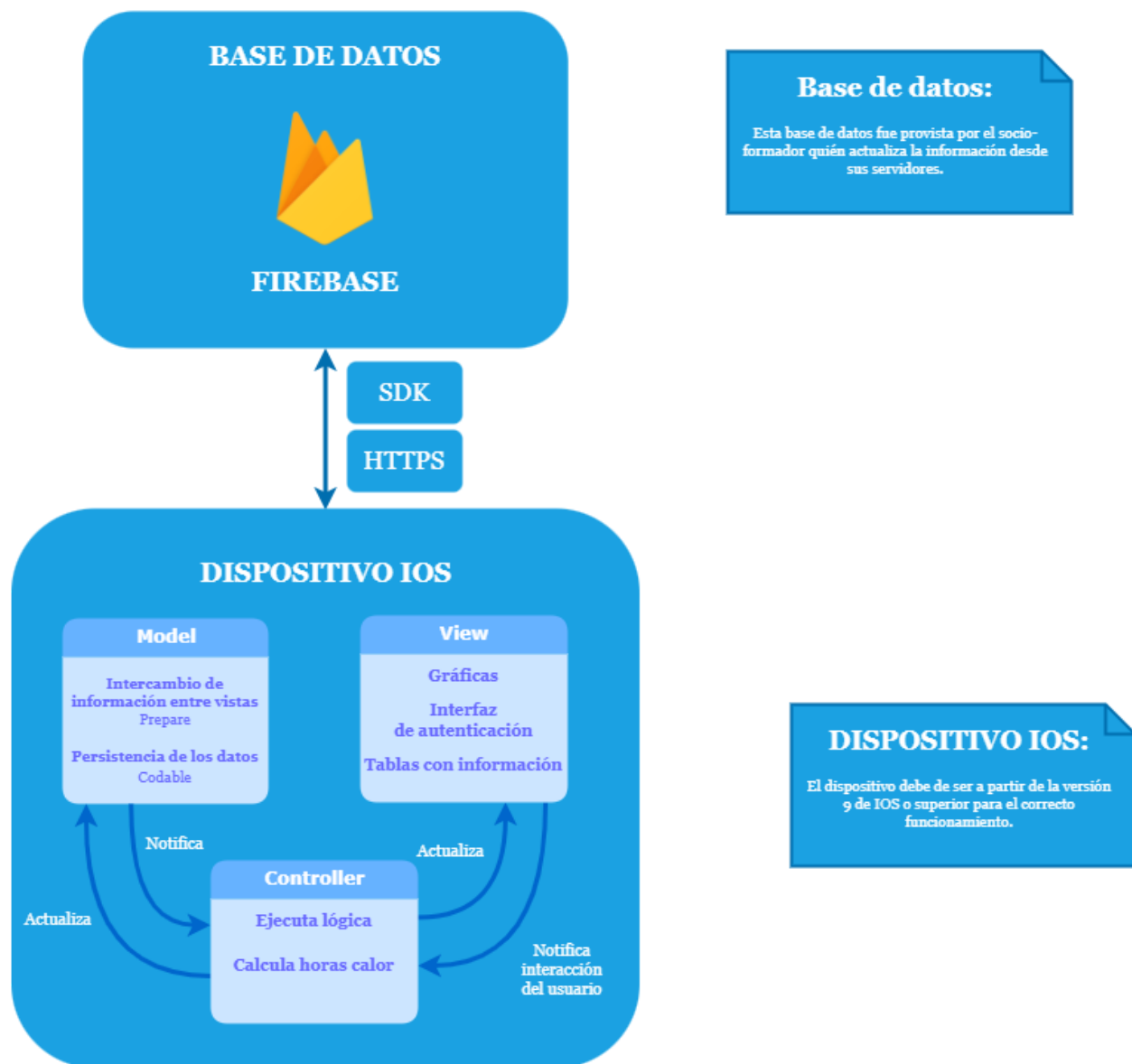
MATRIZ DE RIESGOS						
Nº	Causa	Evento	Descripción	Probabilidad	Impacto	Nivel Riesgo
1	Desestimación en la planeación del proyecto.	Retrasos en el desarrollo de la aplicación.	Se presentan retrasos en la entrega de funcionalidad del software.	4	Alto	
2	Falta de familiarización con herramientas.	Entrega de aplicación deficiente.	Falta de familiarización con las herramientas usadas causa un producto final con bajo estándar.	3	Alto	
3	Falta de seguridad (reglas) en base de datos	Los datos de la base de datos fueron modificados sin permiso.	Un usuario sin autorización obtuvo acceso a la base de datos en Firebase mediante la API y modificó datos sin permiso.	3	Alto	
4	Las reglas de Firestore fueron mal configuradas.	El usuario pudo modificar (escribir y borrar) datos en la base de datos	Debido a que las reglas de la base de datos no estaban bien definidas un usuario tuvo permiso de modificar datos.	2	Alto	

Etapa 2. Diseño

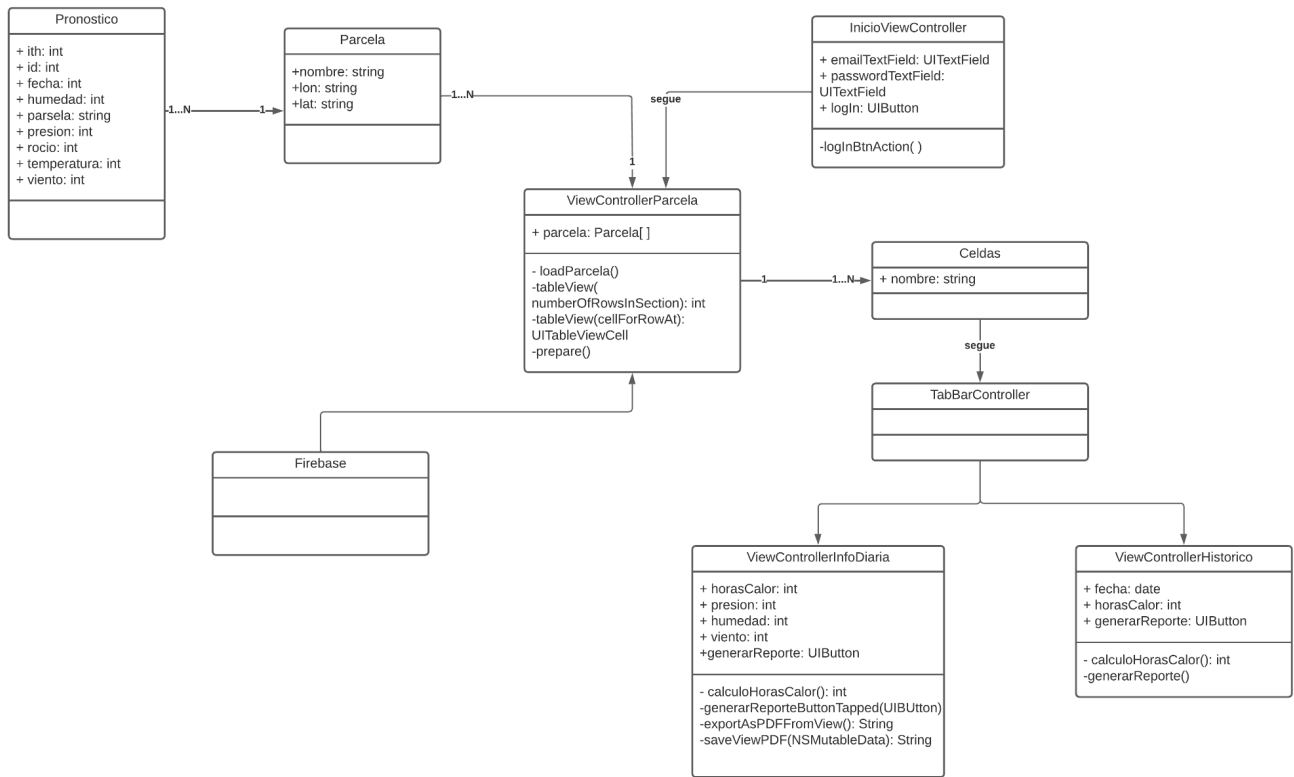
1. Diagrama del diseño de la Base de datos.



2. Diagrama de la arquitectura de la app.



3. Diagrama de clases de la app.



4. Diseño de las interfaces de usuario.

Las siguientes pantallas son prototipos para las interfaces de usuario, todas desarrolladas teniendo en mente que sea fácil de entender y de interactuar para el usuario.



groSky

Inicia Sesión

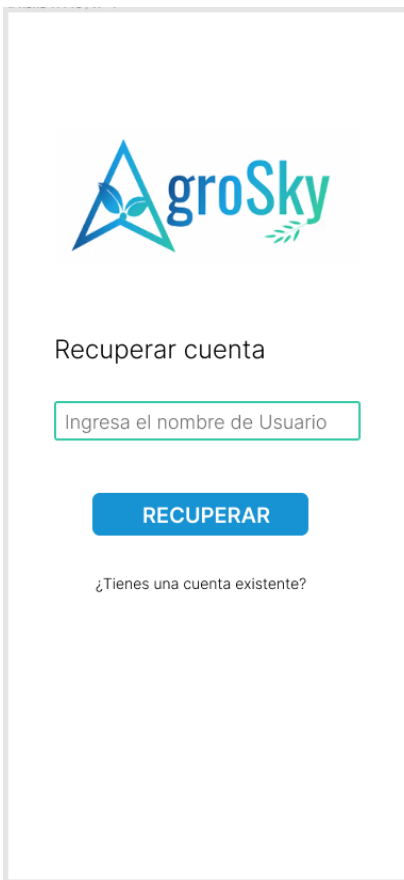
Usuario

Contraseña

INGRESAR

¿Olvidaste tu contraseña?

¿No tienes una cuenta? [Crea una aquí](#)



groSky

Recuperar cuenta

Ingresa el nombre de Usuario

RECUPERAR

¿Tienes una cuenta existente?



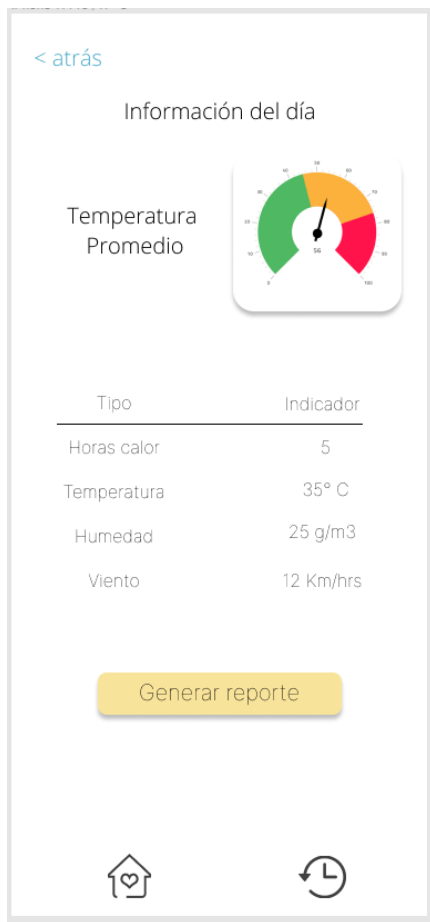
Hecho

Elige el lugar deseado

Rancho Las peñas

El ranchito bonito

Rancho san bartolome



< atrás

Histórico

Fecha	Horas calor
22/Abril/2022	5
21/Abril/2022	3
20/Abril/2022	7
19/Abril/2022	2
18/Abril/2022	4
17/Abril/2022	1

Generar reporte

Home icon, Clock icon

5. Integración de seguridad en sistemas computacionales.

Se trabajará con el protocolo HTTPS. El trabajar con este protocolo permite mayor seguridad al encriptar los mensajes enviados entre el servidor y el cliente al utilizar TLS/SSL, asegurando su integridad mientras estén en tránsito. Su uso permite protección ante ataques de “man-in-the-middle” y contra manipulación de los datos.

Firebase cuenta con sus reglas de seguridad, al estar certificado según los principales estándares de seguridad y privacidad. Cumplen con los procesos de evaluación de ISO y SOC y sus frameworks de Privacy Shield que cumple con reglamentos internacionales. Para acceder a la colección donde se encuentran los datos de la aplicación a esta solo se puede entrar si es un administrador autorizado para poder modificar datos en caso de ser necesario. Un usuario regular de la aplicación no tiene acceso directo a esta información y la puede visualizar mediante la aplicación solamente.

Etapa 3. Desarrollo

1. URL del repositorio de software con el código del servidor, código del cliente.

Repositorio en GitHub: <https://github.com/anguloteresa/AgroskyHorasCalor.git>

2. Configuración de la infraestructura necesaria para desarrollar el proyecto.

El cliente nos proporcionó gran parte de la infraestructura necesaria para el desarrollo del proyecto, por lo que no es necesario instalar o configurar los servidores y redes que hacen parte de la infraestructura del proyecto.

El cliente se encargó de recopilar los datos necesarios para el desarrollo de la solución y los almacenó en un repositorio de Firebase, la única configuración necesaria en este caso sería la unión del repositorio de Firebase con el ambiente de desarrollo usando el lenguaje *Swift*. Para efectuar esta unión es necesario descargar el archivo de configuración proporcionado por Firebase, instalar en el ambiente de desarrollo las dependencias necesarias para trabajar con Firebase y finalmente agregar el código de inicialización también proporcionado por Firebase.

3. Documentación de reuniones con el cliente, desarrolladores y equipo de pruebas.

Para el desarrollo de la aplicación se agendaron tres juntas con el cliente y los desarrolladores, a continuación se muestra cuando se efectuaron dichas juntas, su propósito y su impacto final en el desarrollo.

Fecha	Proposito	Impacto
08/Abr/22	El cliente nos informó acerca de su empresa, su propósito y necesidades actuales, además se presentaron las problemáticas a resolver y se resolvieron dudas iniciales sobre dichas problemáticas.	Se eligió la problemática a resolver de cada equipo, es decir, nuestra futura aplicación.
22/Abr/22	Se resolvieron dudas de acuerdo a los requerimientos de la aplicación, que posibles funcionales debe incluir y se aclararon dudas de la base de datos en Firebase y los datos que dicha herramienta iba a tener.	Se pudo establecer un alcance del proyecto y las tareas a realizar para completar la aplicación.

29/Abr/22	Reunión con el encargado del área de computación de la empresa donde se resolvieron algunas dudas técnicas de la aplicación.	Se pudo iniciar el desarrollo de la aplicación al ya tener muy claro lo que se iba a realizar y lo que se iba a necesitar.
-----------	--	--

4. Identificación de ataques informáticos.

Los ataques informáticos son crímenes cibernéticos al intentar entrar sin autorización a un sistema con el fin de dañar o robar información confidencial. En el caso del reto los delitos a los que se tiene que estar pendiente son principalmente los que caen en la categoría de delitos contra la confidencialidad, integridad y disponibilidad de los datos y sistemas informáticos y los delitos relacionados con la informática. Es importante identificar estos ataques y establecer medidas para evitarlos o disminuir su impacto para garantizar al cliente que sus datos estén protegidos.

- **Hacking:** la entrada no autorizada a sistemas, redes y datos, puede ser simplemente el acceso a información no autorizada o también causar daño a los sistemas atacados.
- **DoS (Denial of Service):** Denegación de servicio, es decir, tiene como objetivo inhabilitar el uso de la aplicación al sobrecargar los servidores con información y prevenir que usuarios reales puedan acceder al sistema.
- **Phishing:** Suplantación de identidad a través de engaños para que el usuario comparta información confidencial, como sus contraseñas.

5. Métodos de protección de ataques informáticos.

En primera instancia se tiene una pantalla de inicio de sesión que verifica la autenticidad del usuario a través de su respectiva contraseña. Esta es una de las medidas más básicas que se pueden y deben de implementar para impedir que cualquier persona acceda a los datos de la aplicación.

Para la conexión de la base de datos con la aplicación, es necesario que cuenten con la misma llave, o la aplicación no puede ser desplegada. Esto previene que las personas que no la conozcan puedan leer, modificar o borrar información de la base de datos en la nube.

Cabe mencionar, que solo los administradores de la aplicación tienen la funcionalidad de crear cuentas de usuario, es decir, un empleado no puede tener acceso a la información a menos de que se le otorguen las credenciales de acceso. Si a un usuario se le llega a olvidar su contraseña, tendría que contactar directamente a los administradores para que se le proporcione una nueva. Al limitar la aplicación a no tener una función de recuperación de contraseña, se evita también que pueda ocurrir algún ataque por phishing, donde se puedan obtener las contraseñas de los usuarios para acceder más tarde a la aplicación.



Inicia Sesión

Correo electrónico

Contraseña

INGRESA

La aplicación evita usar más datos relacionados al usuario, es decir, no pregunta por información que realmente no es indispensable ni relacionada con su correcto funcionamiento. De esta manera hacemos que sea lo menos atractiva posible para que terceras personas no muestren interés por los datos almacenados.

Otro método de protección contra ataques informáticos es que se limita qué individuos tienen acceso a la base de datos y al sistema principal. Solo personal autorizado de parte de DiGiSky tiene acceso a esta información mientras el resto de los usuarios solo pueden visualizar lo que tienen autorizado dentro de la aplicación.

Adicionalmente, los usuarios están limitados a cierta información dentro del sistema, es decir, no todos los usuarios tienen autorización de visualizar la información de todas las parcelas. Esto logra que los usuarios no hagan mal uso de los datos de la aplicación, divulgando o utilizando los mismos con malas intenciones.

Etapa 4. Pruebas

1. Documento de estrategia de pruebas.

Alcance:

Este documento servirá para la documentación, revisión y aprobación de las distintas pruebas a realizar en nuestra aplicación, el equipo de desarrolladores como conjunto se encargará de realizar estas actividades. Las pruebas se van a realizar conforme al cronograma propuesto en la *Etapa 1: Requerimientos* como se observa en el diagrama las pruebas se van a realizar de la mano del desarrollo de la aplicación.

Fecha	Tipo de prueba
30/Abr/2022	Pruebas unitarias
02/May/2022	Pruebas de integración
03/May/2022	Pruebas de sistema y regresión
04/May/2022	Pruebas funcionales
05/May/2022	Pruebas de aceptación/alfa.

Enfoque de pruebas:

Debido al corto tiempo para el desarrollo de la aplicación las pruebas se van a emplear de manera manual y los desarrolladores van a ser los encargados de realizarlas. Como se observó en el punto anterior se van a incluir pruebas en distintos niveles del desarrollo:

- Pruebas unitarias, se revisan los componentes uno por uno y los encargados son los desarrolladores.
- Pruebas de integración, se integran los componentes y se prueba su funcionalidad como un conjunto.
- Pruebas de sistema, una prueba preliminar del sistema completo donde se evalúan los elementos funcionales y no funcionales.
- Pruebas de aceptación, se hacen pruebas de los requerimientos antes propuestos.

De los tipos de pruebas que se van a aplicar en los distintos niveles de prueba tenemos las pruebas de seguridad, pruebas funcionales, pruebas de regresión, pruebas alfa y pruebas no-funcionales. Como se mencionó anteriormente estas pruebas van a ser realizadas manualmente por los desarrolladores.

Ambiente de pruebas:

Las pruebas van a ser realizadas en el ambiente de desarrollo de *Xcode*, por lo que será necesario contar con un equipo de computo con sistema operativo macOS.

Herramientas para pruebas:

Por el marco de tiempo y falta de experiencia no se va a utilizar software que automatice las pruebas.

Control de versiones:

Fecha	Versión/Modificaciones	Ejecución de pruebas
02/May/2022	V1.0 - Aplicación base con navegación entre pantallas	Se probó la correcta implementación de la navegación mediante pruebas de integración.
02/May/2022	V1.1 - Conexión con base de datos y recopilación de información en ella. - Visualización de información (parcelas) en pantalla. - Botón para exportar el documento en formato PDF añadido.	Se probó la conexión a la base de datos y si los datos en pantalla estaban correctos y corresponden los los Documentos en la Colección de <i>Parcelas</i> de <i>Firebase</i> . Pruebas de sistema y regresión
03/May/2022	V2.0 - Autenticación de credenciales en inicio de sesión. - Exportación de documento en formato PDF de reporte generado con las horas calor. - Visualización en tablas y gráfica de las horas calor del día. - Visualización de las horas calor de cada día en una tabla en la pantalla <i>Historico</i> .	Se revisó que la autenticación funcionará correctamente, los datos en las tablas y gráfica coincidieran con la información proporcionada por el cliente en la base de datos y que se exportara de manera correcta el reporte generado dentro de la aplicación. Pruebas funcionales, de integración y regresión.
04/May/2022	V3.0 - Cambios en la visualización de datos en pantalla histórico.	Se hicieron pruebas funcionales y de integración enfocadas en la pantalla de Histórico donde se visualizan los datos de horas calor de cada día.
05/May/2022	V4.0 - Botón en pantalla de inicio redondeado.	Pruebas de sistema, aceptación/alfa.

	<ul style="list-style-type: none"> - Tabla de pantalla histórico con datos ordenados de menor a mayor por las fechas. - Exportación del reporte generado en la pantalla de histórico. 	
--	---	--

Análisis de riesgo:

Riesgo	Medida de contingencia
El tiempo para desarrollo y pruebas es muy corto	Dar prioridad a cada actividad y repartir el trabajo
Falta de familiarización con las herramientas y plataformas usadas.	Buscar apoyo con el encargado del proyecto (profesor) en caso que sea necesario.
Falta de pruebas automatizadas	Realizar pruebas manuales exhaustivas en cada componente y su integración en el sistema.

Revisiones y aprobaciones:

Fecha	Versión	Autor(es)	Autorizó
02/May/2022	V1.0	Equipo 4	Todo el equipo
02/May/2022	V1.1	Equipo 4	Todo el equipo
03/May/2022	V2.0	Equipo 4	Todo el equipo
04/May/2022	V3.0	Equipo 4	Todo el equipo
05/May/2022	V4.0	Equipo 4	Todo el equipo

2. Plan de pruebas.

Paso 1. Análisis del producto:

Usuario de la aplicación: La aplicación será usada por los clientes de nuestro cliente, DiGiSKY, que hayan contratado el servicio de AgroSky para sus parcelas agrícolas.

Propósito de la aplicación: La aplicación le permitirá al cliente ver las horas calor e información relacionada con las horas calor de su/s parcela/s, también se podrá generar un reporte en formato PDF con la información antes mencionada.

Software/Hardware relacionado: La aplicación fue desarrollada con lenguaje de programación *Swift*, se usó Xcode como ambiente de desarrollo y los datos usados en la aplicación están almacenados en una base de datos de Firebase. El producto final es una aplicación para dispositivos móviles que cuentan con iOS.

Paso 2. Estrategia de pruebas:

El [documento de estrategia](#) de pruebas se encuentra en este mismo documento y es el punto anterior al plan de pruebas.

Paso 3. Objetivo de pruebas:

El objetivo de nuestro plan de pruebas es asegurar que los requerimientos funcionales y no funcionales antes propuestos en la Etapa 1: Requerimientos sean cumplidos de manera satisfactoria. También es necesario que la aplicación sea usable y que todas las funcionalidades propuestas sean convenientes para el cliente y vayan a ser usadas.

Paso 4. Criterios de pruebas:

Los criterios de prueba nos proporcionan un estándar o regla a efectuar al momento de estar realizando el procedimiento de pruebas, para las pruebas se van a tener dos criterios, los cuales son de suspensión y salida. Como lo dice su nombre los criterios de suspensión nos indican cuándo o bajo qué condición se van a suspender las pruebas y los criterios de salida especifican los criterios que hacen que un proceso de pruebas sea exitoso.

Criterio de Suspensión	Criterio de salida
Los cambios no funcionan para uno o más miembros del equipo.	Las pruebas y la tasa de ejecución son 100% exitosas.
La tasa de ejecución es igual o menor al 70%	Los cambios funcionan de manera exitosa para todos los miembros de equipo.
Si los cambios que generan errores no han sido corregidos	La tasa de aprobación (casos de prueba pasados / casos de prueba ejecutados) es mayor o igual a 90%

Paso 5. Planeamiento de recursos:

A continuación se tiene un resumen de los recursos necesarios para el desarrollo del proyecto y sus pruebas correspondientes.

Recursos Humanos

No.	Miembro	Tareas
1	Desarrollador de equipo 4	Ejecutar las pruebas. Implementar y probar casos de prueba. Reportar errores.

Recursos de sistema

No.	Recurso	Descripción / Uso
1	Simulador movil	El simulador nos servirá para ver el funcionamiento de la aplicación.
2	Computadora	Corregir los errores recopilados en las pruebas y corregirlos. Ejecutar el simulador y la aplicación.

Paso 6. Ambiente de plan de pruebas:

Nuestro ambiente de pruebas incluye el IDE Xcode, la plataforma de Firebase y el simulador de dispositivo móvil con iOS.

Paso 7. Calendario y estimación:

Tarea	Miembro	Esfuerzo Estimado	Fecha
Creación de especificación de pruebas	Equipo 4	10 hora-persona	04/05/2022 - 05/05/2022
Ejecución de pruebas	Equipo 4	60 hora-persona	30/05/2022 - 05/05/2022
Reporte de pruebas	Equipo 4	4 hora-persona	04/05/2022 - 05/05/2022
TOTAL		74 hora-persona	

Paso 8. Entregables de pruebas:

Los entregables de pruebas son los siguientes:

- Casos de prueba.
- RTM (Matriz de rastreamiento de requerimientos).
- Reporte de pruebas.
- Informe de cierre de pruebas.
- Reporte de incidentes.

Dichos entregables y su información correspondiente se encuentran a lo largo de este documento.

3. Casos de prueba y datos de prueba

01. Pruebas en Inicio de sesión	
ID TC.	Caso a probar
TC1001	La aplicación autentica si las credenciales proporcionadas por el usuario se encuentran en la herramienta de autenticación de Firebase.
TC1002	La aplicación despliega un mensaje en caso de que las credenciales sean incorrectas.
TC1003	La aplicación permite el acceso a la siguiente pantalla cuando el usuario fue autenticado.

02. Pruebas en la visualización de horas calor	
ID TC.	Caso a probar
TC2001	El usuario puede seleccionar una parcela y cambiar de parcela.
TC2002	La aplicación muestra las horas calor de la parcela seleccionada mediante una tabla.
TC2003	El sistema despliega los últimos datos capturados en la base de datos.

03. Pruebas en la generación de reportes	
ID TC.	Caso a probar
TC3001	La aplicación genera un reporte donde se especifican las horas calor y el día que corresponde a la cantidad de horas calor.
TC3002	La aplicación le permite al usuario seleccionar un periodo de tiempo para visualizar el reporte de horas calor.
TC3003	La información y los datos desplegados en el reporte son correctos y corresponden a la información en la base de datos.

04. Pruebas en exportación de reportes	
ID TC.	Caso a probar
TC4001	El documento exportado está en formato PDF.
TC4002	El usuario puede guardar el reporte en PDF a su dispositivo.
TC4003	El usuario puede cancelar la exportación del reporte.

4. RTM (Requirement Traceability Matrix)

Requerimiento	Descripción req.	ID. Caso de prueba	Estatus
RF01000	Inicio de sesión a la aplicación.	TC1001, TC1002, TC1003	TC1001 - Pasa TC1002 - Pasa TC1003 - Pasa
RF02000	Visualización de horas calor en la aplicación.	TC2001, TC2002, TC2003	TC2001 - Pasa TC2002 - Pasa TC2003 - Pasa
RF03000	Generación de reportes	TC3001, TC3002, TC3003	TC3001 - Pasa TC3002 - Pasa TC3003 - Pasa
RF04000	Exportación de reportes.	TC4001, TC4002, TC4003	TC4001 - Pasa TC4002 - Pasa TC4003 - Pasa

5. Reporte con resumen y resultados de las pruebas.

Reporte de pruebas						
Pruebas empleadas	Integration Test	Unit Test	System Test	Regresion Test		
EJECUTADAS	PASADOS			12		
	FALLADOS			0		
	(Total) PRUEBAS EJECUTADAS				12	
PENDIENTES					0	
EN PROGRESO					0	
BLOQUEADOS					0	
	(Sub-Total) PRUEBAS PLANEADAS				12	
	(PENDIENTES + EN PROGRESO + BLOQUEADAS + EJECUTADAS)					
Funcion	Descripción	% CPs Ejecutados	% CPs Pasados	CPs pendientes	Prioridad	Comentarios
Inciar sesión	Inicio de sesión en la aplicación	100%	100%	0	High	
Ver horas calor	Visualización de horas calor	100%	100%	0	High	
Horas calor diarias	Ver las horas calor del día e información pertinente	100%	100%	0	High	
Autenticación de usuario	Autentica el usuario al iniciar sesión	100%	100%	0	High	
Generación de reporte	Genera un reporte con las horas calor de cada día.	100%	100%	0	High	
Exportar el reporte	Permite exportar el reporte generado en formato PDF.	100%	100%	0	High	
Elegir marco de tiempo	Se puede elegir un marco de tiempo para visualizar el reporte de horas calor.	100%	100%	0	High	

6. Informe de cierre de las pruebas.

Resumen de proyecto: La aplicación busca satisfacer la necesidad del cliente (DiGiSKY) y de los usuarios de la herramienta agrícola (AgroSky) desarrollada por el cliente. Mediante la recopilación de información relacionada a las horas calor de una parcela, la aplicación se muestra la información procesada en tablas por distintos marcos de tiempo, elegidos por el usuario, y además se pueden realizar reportes indicando las horas calor por día en un marco de tiempo determinado por el mismo usuario, finalmente este reporte se puede exportar en formato PDF para ser compartido o guardado en el dispositivo.

Ambiente de pruebas y desarrollo: La aplicación fue desarrollada para dispositivos iOS y las pruebas se realizaron en dispositivos móviles con este sistema operativo.

Pruebas realizadas (Alcance): Se realizaron pruebas siguiendo los requerimientos funcionales antes propuestos, durante el desarrollo se realizaron pruebas unitarias, integración, de regresión y alfa.

Pruebas no realizadas (Fuera de alcance): Debido al tiempo limitado para desarrollo no se realizaron algunas pruebas o áreas de la aplicación como lo son:

- Compatibilidad con otros dispositivos (*iPad, iPod*).
- Pruebas automatizadas.
- Visualización y despliegue en la *App Store*.


Detalles de ejecución de pruebas: Para información detallada de las pruebas realizadas, su función, si fueron exitosas o no, ver el punto [5. Reporte de pruebas](#) de este documento.

Comentarios Equipo Pruebas: La aplicación tiene áreas de mejora y posibles funcionalidades que la harían una aplicación más completa y le darían una mejor experiencia al usuario, pero con el limitado tiempo para el desarrollo se logró un buen resultado que cumple con las funcionalidades propuestas en la etapa de requerimientos.

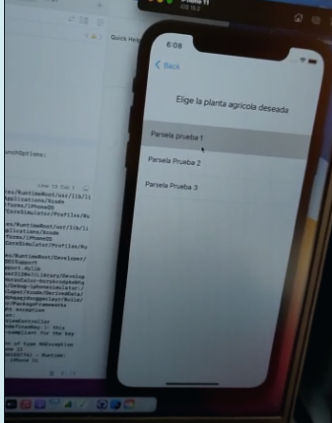
Conclusión: La aplicación actual es estable.

7. Reporte de incidentes.

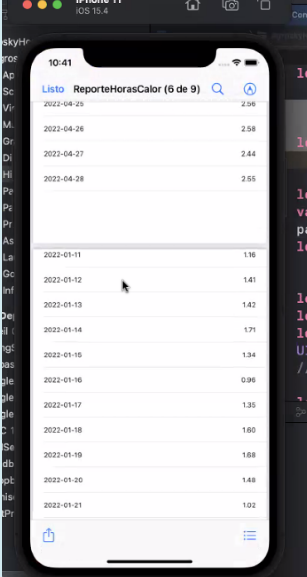
ID Incidente	IR1000
Nombre proyecto	AgroSky App 2: Horas calor
Nombre módulo	ViewController, página inicio sesión
Fase de prueba	Pruebas de sistema
Tipo de error	Funcional

Impacto	Alto
Prioridad	Alta
Resumen	Al hacer click al botón de ingresar no manda a la siguiente pantalla.
Descripción	<p>En la pantalla de iniciar sesión al hacer click en el botón de iniciar sesión no manda a la siguiente pantalla aun cuando los datos están autenticados y correctos.</p> 
Estatus	Resuelto y cerrado

ID Incidente	IR2000
Nombre proyecto	AgroSky App 2: Horas calor
Nombre módulo	DiaTableViewController, página de selección de parcela
Fase de prueba	Pruebas de sistema
Tipo de error	Funcional
Impacto	Alto
Prioridad	Alta
Resumen	Al hacer clic en la celda de una parcela se termina la ejecución por un

	error.
Descripción	<p>En la pantalla donde se despliegan las parcelas al hacer clic en una celda manda una excepción por una llave no definida dentro del controlador DiaTableViewController.</p> <pre>Terminating app due to uncaught exception 'NSUnknownKeyException', reason: '[<AgroskyHorasCalor.DiaTableViewController 0x15f20a6c0> setValue:forUndefinedKey:]: this class is not key value coding-compliant for the key exportButton.'</pre> 
Estatus	Resuelto y cerrado

ID Incidente	IR3000
Nombre proyecto	AgroSky App 2: Horas calor
Nombre módulo	HistoricoViewController, página generación de reporte histórico
Fase de prueba	Pruebas de sistema
Tipo de error	Funcional
Impacto	Medio
Prioridad	Media
Resumen	Al exportar el reporte histórico en un documento PDF no se ven todos los datos
Descripción	En la pantalla donde se despliegan los datos de horas calor por día al exportar los datos en un documento PDF se cortan algunos datos

	
Estatus	Resuelto y cerrado

Etapas 5. Despliegue

1. Planeación de la migración del software.

La planeación de la migración del software nos permite, migrar aplicaciones existentes en sistemas operativos o dispositivos antiguos a plataformas más modernas como lo puede ser un sistema operativo virtual que facilite el desarrollo y las pruebas en la aplicación. Este paso no fue desarrollado en nuestra solución, ya que la infraestructura computacional fue proporcionada por el cliente y esta misma ya estaba validada y monitoreada por el cliente. Los servicios elegidos por el cliente para el almacenamiento de datos, *Firebase*, nos asegura una alta calidad en los servicios que esta misma herramienta proporciona.

2. Manual de usuario.

Requisitos.

- Para el uso de la aplicación es necesario contar con un dispositivo iOS.
- Contar con una cuenta de usuario (correo electrónico y contraseña) proporcionada por DiGiSKY o AgroSky.

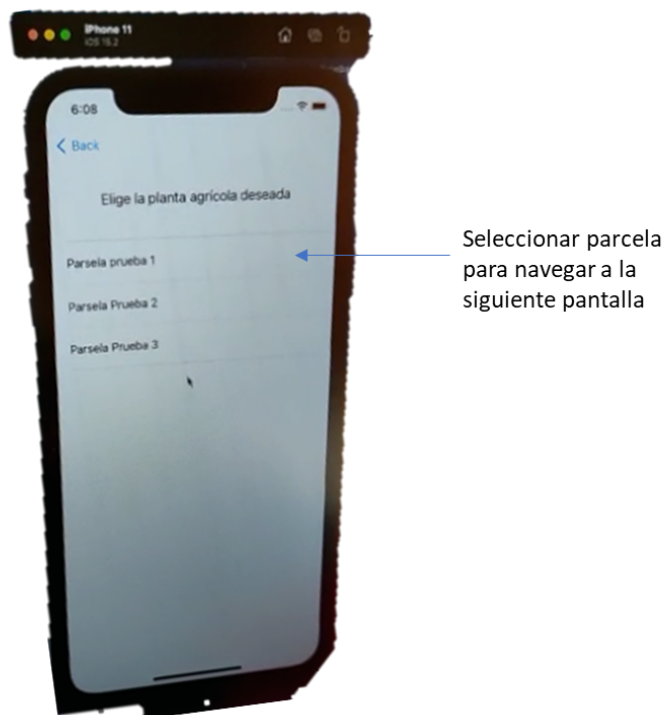
Uso de la aplicación.

Para hacer uso de la aplicación es necesario tener las credenciales mencionadas en los requisitos.

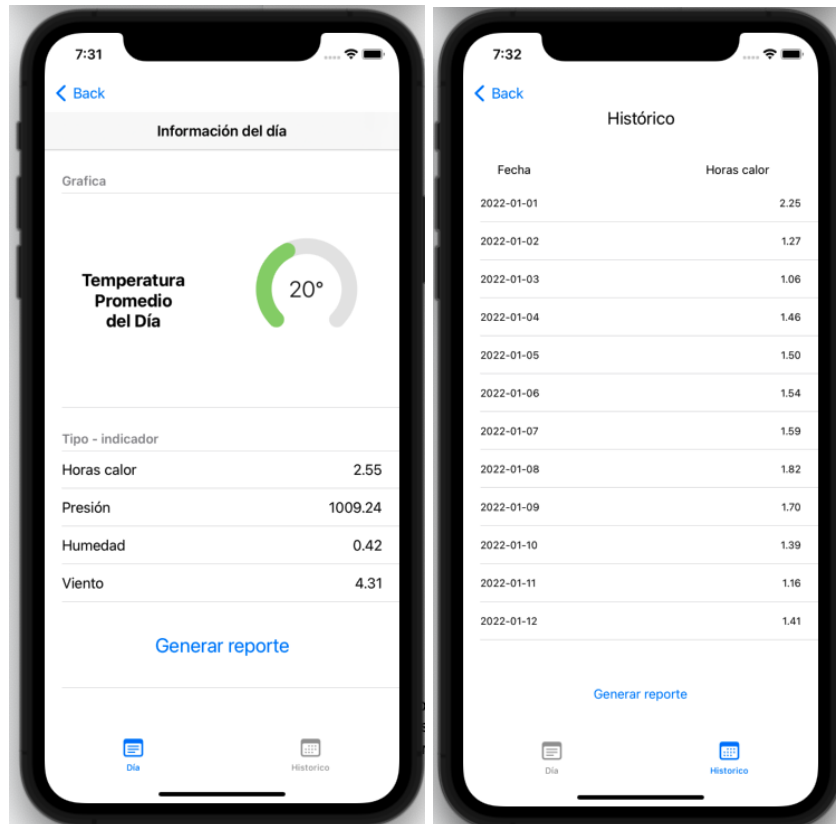
La primera pantalla permite ingresar dichas credenciales en caso de que sean correctas se navega a la siguiente pantalla de lo contrario se muestra un error indicando que uno o ambos datos ingresados son incorrectos.



La siguiente pantalla nos muestra las parcelas agrícolas y al seleccionar una parcela se navega a la siguiente pantalla.

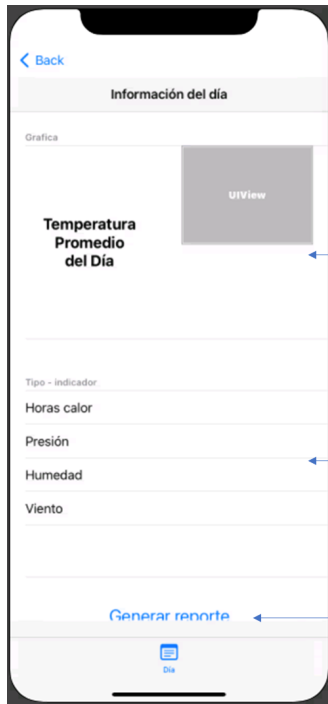


En esta pantalla, en la parte inferior, se pueden elegir dos opciones, la primera, el reporte y gráfica de horas calor del día o la segunda, el reporte de las horas calor de manera histórica con su correspondiente día.



En ambas opciones hay un botón con el texto “Generar reporte” con el cual se abre un diálogo donde se le dan opciones al usuario para ver que quiere hacer con el reporte generado (copiar, guardar a dispositivo, cancelar acción), este reporte se guarda como un documento PDF.

Opción 1

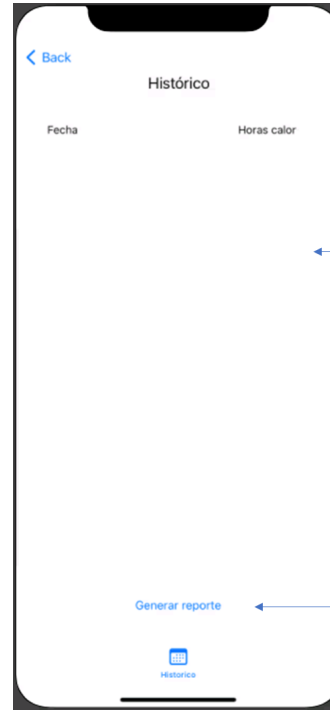


En esta sección se visualiza la grafica

En esta sección se visualizan los datos

Botón para generar reporte

Opción 2



En esta sección se visualizan los datos

Botón para generar reporte

Características.

- Navegación entre pantallas intuitiva.
- Seguridad y autenticación de datos.
- Información importante de las parcelas.

Forma de uso.

La aplicación cuenta con una navegación entre pantallas mediante el uso de botones y celdas a elegir por el usuario.

3. Manual de instalación

Para desarrolladores:

- Es necesario contar con una computadora Mac o algún dispositivo que permita trabajar con el lenguaje de programación Swift (iOS) y el entorno de desarrollo XCode 12 en adelante.

- De igual manera, es necesario poder acceder al repositorio en GitHub (<https://github.com/anguloteresa/AgroskyHorasCalor.git>) y clonarlo en el respectivo dispositivo para tener acceso a todos elementos y códigos del proyecto (aplicación).

Para usuarios:

- No se pudo realizar un archivo ejecutable o publicar a la App Store por el marco de tiempo.

4. Capacitación.

La capacitación a usuarios no pudo ser efectuada por el marco de tiempo para desarrollo, pero se pudo realizar una demostración del funcionamiento de la aplicación con el cliente.

5. Publicación de la app.

La aplicación no pudo ser publicada en la App Store por la falta de tiempo presente, se sabe que la publicación y verificación de una aplicación puede durar dos semanas o más para estar disponible al público en la App Store, este tiempo representa casi todo el tiempo que tuvimos para el desarrollo de la solución.