

Universidad Autónoma Metropolitana
Unidad Azcapotzalco

DIVISIÓN DE CIENCIAS BÁSICAS E INGENIERÍA
LICENCIATURA EN INGENIERÍA EN COMPUTACIÓN

Sistema para la determinación del espacio entre semillas para su
siembra en un terreno.



Modalidad: Proyecto Tecnológico
Primera versión
Trimestre 2017 Primavera

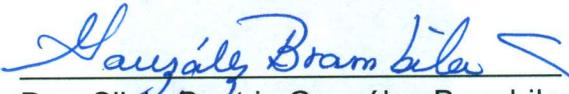
Sonia Francisco Atzin
2171998960
al2171998960@azc.uam.mx

Asesora: Dra. Silvia Beatriz González Brambila
Profesora Titular
Departamento de Sistemas
sgb@correo.azc.uam.mx

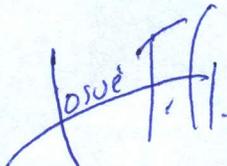
Coasesor: Mtro. Josué Figueroa Gonzáles
Profesor Asociado
Departamento de Sistemas
jfgo@correo.azc.uam.mx

21 de julio de 2017

Yo, Silvia Beatriz González Brambila, declaro que aprobé el contenido del presente Reporte de Proyecto de Integración y doy mi autorización para su publicación en la Biblioteca Digital, así como en el repositorio Institucional de la UAM Azcapotzalco.


Dra. Silvia Beatriz González Brambila

Yo, Josué Figueroa González, declaro que aprobé el contenido del presente Reporte de Proyecto de Integración y doy mi autorización para su publicación en la Biblioteca Digital, así como en el repositorio Institucional de la UAM Azcapotzalco.



Mtro. Josué Figueroa González

Yo, Sonia Francisco Atzin, doy mi autorización a la Coordinación de Servicios de Información de la Universidad Autónoma Metropolitana, Unidad Azcapotzalco, para publicar el presente documento a la Biblioteca Digital, así como en el Repositorio Institucional de UAM Azcapotzalco.



Sonia Francisco Atzin

Resumen

El trabajo presentado a continuación tiene como objetivo fundamental apoyar a usuarios que estén interesados en realizar una siembra en un determinado terreno, para ello se ha realizado una aplicación móvil que permite determinar el área de siembra de acuerdo a coordenadas geográficas obtenidas con el GPS del dispositivo móvil en donde se encuentra la aplicación instalada.

Una vez que se tiene establecida el área a sembrar, la aplicación indica la distancia que debe existir entre las semillas del producto seleccionado. La aplicación se realizó en el entorno de desarrollo de Android Studio con las bibliotecas necesarias para el desarrollo para dispositivos móviles, una de las tecnologías más utilizadas actualmente por gran cantidad de personas, razón por la que se eligió esta plataforma para el desarrollo.

La realización de este proyecto implicó el uso de conocimientos relacionados con Diseño de Sistemas, considerando funcionalidades e interfaces intuitivas. También involucró conocimientos de Bases de Datos y de programación, además de otros conocimientos como matemáticas para el manejo de coordenadas y acceso a los sistemas de un dispositivo móvil. Si bien significó un reto debido a la poca experiencia con la que se contaba, durante el desarrollo del proyecto se lograron integrar todos estos conocimientos llegando a un resultado adecuado.

Tabla de contenido

Resumen	3
1. Introducción	6
2. Antecedentes.....	7
3.1 Proyectos de integración o terminales	7
3.2 Artículos	7
3.3 Software	8
3. Justificación	9
4. Objetivo General.....	10
5.1 Objetivos Específicos.....	10
5. Marco Teórico.....	11
6. Desarrollo del Proyecto.....	14
7.1 Tablas.....	15
7. Resultados.....	21
Las capturas de pantalla correspondientes a las pruebas realizadas se presentan en las Figuras de la 22 a la 31.....	22
8. Conclusiones	27
9. Bibliografía.....	28
10. Entregables	30

Tabla de Figuras

Figura 1. Fórmula de Haversine	11
Figura 2. Módulos Componentes del Sistema	14
Figura 3. Diagrama Entidad Relación de Proyecto.....	14
Figura 4. Caso de Uso: Agregar Terreno.	16
Figura 5. Diagrama de Secuencia en Caso de Éxito.	16
Figura 6. Diagrama de Secuencia en Caso de Fracaso.	17
Figura 7. Inicio de la aplicación móvil.	18
Figura 8. Punto de partida para datos del terreno.	18
Figura 9. Segundo punto considerado para obtener un lado.	18
Figura 10. Último punto necesario para calcular el área del terreno.....	18
Figura 11. Permite volver a medir o continuar con el proceso de siembra.....	18
Figura 12. Al continuar se ingresan el nombre, los pasos y el producto a sembrar (aguacate).....	18
Figura 13. Registro de terreno con nombre, pasos dados y producto a sembrar (Maíz).	19
Figura 14. Registro de terreno con nombre, pasos dados y producto a sembrar (Naranja). 19	
Figura 15. Detalles del terreno, dimensiones, área, etc.....	19
Figura 16. Seleccionar la forma de sembrar.....	19
Figura 17. Indicaciones de siembra.....	19

Figura 18. Botón Cargar para visualizar los terrenos guardados.....	19
Figura 19. Lista de terrenos guardados.....	20
Figura 20. Seleccionar producto a sembrar.....	20
Figura 21. Indicaciones de cómo realizar la siembra.....	20
Figura 22. Resultados de Prueba01.....	22
Figura 23. Resultados de Prueba02.....	22
Figura 24. Resultados de Prueba03.....	23
Figura 25. Resultados de Prueba04.....	23
Figura 26. Resultados de Prueba05.....	24
Figura 27. Resultados de Prueba06.....	24
Figura 28. Resultados de Prueba07.....	25
Figura 29. Resultados de Prueba08.....	25
Figura 30. Resultados de Prueba09.....	26
Figura 31. Resultados de Prueba10.....	26

Tablas

Tabla 1. Productos de Prueba.....	12
Tabla 2. Bibliotecas Implementadas.....	12
Tabla 3. Tablas que componen la Base de Datos utilizada en el proyecto.	15
Tabla 4. Terreno	15
Tabla 5. Producto.....	15
Tabla 6. Resultados de Pruebas Realizadas.....	21

1. Introducción

Cuando se siembra es importante determinar la distancia entre cada semilla dadas las dimensiones del terreno, normalmente esto se realiza en dos pasos: el deslinde, que determina los límites del terreno y el fraccionamiento que consiste en determinar los segmentos de terreno utilizados para colocar una semilla, es decir, el espacio que habrá entre esta y otras semillas.

Es importante decir que los segmentos de siembra utilizados varían de acuerdo al producto y a la forma del terreno, existe una gran variedad de estudios que demuestran que se puede incrementar la productividad del terreno a partir de una distribución adecuada de las semillas a plantar [1].

En la actualidad ha resultado compleja la partición de terrenos de siembra, debido a que esta labor toma mucho tiempo, ya que se deben medir diferentes distancias cada vez que coloca una semilla en la posición calculada.

En los terrenos donde se realizan rotaciones de cultivo, las particiones del terreno deben modificarse cada que se siembre un producto diferente para aprovechar el área de siembra de la mejor manera y evitar la propagación de plagas. Por lo tanto, al escoger un producto nuevo, la partición debe variar.

El uso de nuevas tecnologías permite el mejor control de recursos, las personas que se dedican a sembrar intentan optimizar el espacio de sus tierras para poder producir la mayor cantidad posible de productos, sin embargo, no siempre obtienen buenos resultados, ya que en ocasiones basan su siembra en mediciones basadas por su experiencia y no cuentan con medidas precisas del terreno así como dimensiones requeridas por cada producto para su crecimiento adecuado.

En este proyecto, se propone construir una aplicación móvil que permita determinar el espacio en el que se distribuirán las semillas de un cierto producto para que crezcan y produzcan de la mejor manera posible mediante el previo cálculo del área del terreno a sembrar.

2. Antecedentes

3.1 Proyectos de integración o terminales

1. Transmisión y registro de las coordenadas geográficas de un dispositivo móvil [2].

Se trata de una aplicación para un dispositivo móvil, que permite obtener su ubicación en coordenadas geográficas y transmitirlas a un servidor web cada determinado tiempo. Es similar en la presente propuesta ya que de igual forma mediante el GPS se obtendrá coordenadas de un terreno y serán presentadas de manera gráfica en una aplicación móvil.

2. Asignación óptima para el uso de laboratorios y talleres de la División de Ciencias Básicas e Ingeniería [3].

El objetivo general es similar, pues la función principal es la optimización, dicha propuesta optimiza tiempos para el uso de los laboratorios y talleres de la División de CBI, mientras que la propuesta planteada requiere optimizar espacios físicos de terrenos para la siembra de semillas.

3. Sistema de Información Web para la Geolocalización de Unidades Aéreas [4].

Este proyecto utiliza herramientas que permiten conocer la posición geográfica en tiempo real de unidades móviles, unidades aéreas y de personas, la similitud con la propuesta planteada es que se utilizará una herramienta que permita conocer la geolocalización de personas para poder realizar la siembra en las zonas marcadas en el mapa generado al realizar el cálculo del área del terreno.

3.2 Artículos

4. Cultivo de semillas [5].

Es un estudio donde se presentan entre otros, los resultados del registro del área que requieren algunas semillas para su óptimo desarrollo, se asemeja con esta propuesta, ya que de igual forma se asignará mediante la aplicación un área cada semilla para alcanzar un correcto desarrollo, los datos recabados durante este estudio serán utilizados como referencia para el desarrollo de esta propuesta.

3.3 Software

5. Mide Mapas [6].

Aplicación Móvil que mide distancias reales desde *Google Maps*, calcular distancias, perímetros y superficies y los datos almacenados de dicho cálculo los presentan en un mapa, su similitud con esta propuesta en la forma en la que hace el cálculo de las superficies y la forma en que representa los datos en un mapa además del uso del GPS que facilitan la ubicación de los sitios a medir.

6. Syngenta Soluciones [7].

Aplicación móvil que cuenta con una lista de productos agrícolas y se enfoca en dar información especializada de acuerdo con cada producto para evitar plagas, la similitud más importante es que en esta propuesta se hará una lista de semillas similar y de igual forma habrá información especial de cada semilla, sin embargo, sólo será información sobre el diámetro que necesita cada semilla para crecer correctamente.

3. Justificación

Durante el proceso de siembra un factor importante a considerar es la densidad de la siembra, y esta se refiere a la cantidad de semillas que se siembran en un terreno, este factor afecta el rendimiento de las semillas y su crecimiento ya que se vuelven más propensas a enfermedades y proliferación de plagas.

El ingeniero agrónomo Gustavo Ferraris menciona que una siembra correcta, “maximiza la intercepción de radiación fotosintéticamente activa durante el periodo crítico para la definición del rendimiento” [8].

El proyecto planteado tiene como objetivo aplicar la tecnología para resolver el problema de distribuir las semillas de un producto en un terreno de siembra de tal forma que se permita un crecimiento adecuado de los productos.

4. Objetivo General

Desarrollar e implementar una aplicación que permita determinar el espacio entre semillas dentro de un terreno para la siembra.

5.1 Objetivos Específicos

- Diseñar e implementar un módulo que permita obtener información sobre un terreno mediante sus coordenadas.
- Diseñar y desarrollar un módulo que permita al usuario elegir uno de los terrenos y el tipo de producto a sembrar, previamente registrados.
- Diseñar e implementar un módulo que presente un mapa con las divisiones y la ubicación más adecuada en donde se colocarán las semillas a sembrar.
- Diseñar e implementar un módulo que le indique al usuario cuando se encuentra en el punto en donde se debe colocar la semilla.

5. Marco Teórico

Para entender el desarrollo y funcionamiento del proyecto será necesario conocer algunos conceptos particulares, es indispensable conocer que una coordenada se refiere a un eje o línea utilizada para poder determinar la posición de un punto en el espacio, en este caso serán utilizadas para determinar el área de un cuadrilátero, para esto se utilizó el entorno de desarrollo de Android Studio el cual es un software que proporciona herramientas más rápidas para crear todo tipo de aplicaciones en dispositivos Android, así como depuración, edición de código, herramientas para mejorar el rendimiento entre otras cosas, mientras que la base de datos se realizó en SQL (Structured Query Language), SQL es un lenguaje que permite trabajar con base de datos relacionales.

Para el cálculo del área del terreno se utilizará la fórmula de Haversine, con esta fórmula se realiza el cálculo de la distancia de círculo máximo entre dos puntos de un globo teniendo conocimiento de su longitud y latitud. A continuación, se muestra la fórmula Haversine.

$$d = 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right)$$

Figura 1. Fórmula de Haversine

Para que la aplicación pueda funcionar correctamente es necesario ubicarse en un punto de partida y moverse a un punto de llegada de esta manera la aplicación podrá calcular el área, es necesario proporcionar esta información de la manera antes mencionada ya que de lo contrario será imposible que la aplicación pueda hacer cálculo alguno.

El proyecto presentado es de importancia ya que una incorrecta densidad de siembra dentro de un terreno es un problema común y puede afectar el producto del árbol, así como su crecimiento y la calidad de las frutas o verduras. En esta versión de desarrollo se utilizaron tres productos para hacer las pruebas y conocer la densidad de siembra adecuada según los cálculos de las áreas de los diferentes terrenos que se midieron. Los productos de prueba utilizados como referencia se presentan en la Tabla 1.

Tabla 1. Productos de Prueba		
PRODUCTO	DISTANCIA	IMAGEN
Maíz [9]	75 cm	
Naranja [10]	500 cm	
Aguacate [11]	600 cm	

El desarrollo del proyecto se realizó con la implementación de librerías de compatibilidad de Android Studio para dispositivos con una versión de Android menor a 5.0 (Lollipop), las bibliotecas utilizadas para el desarrollo de la aplicación se presentan en la Tabla 2.

Tabla 2. Bibliotecas Implementadas	
NOMBRE	DESCRIPCIÓN
com.android.support: appcompat	Incluye compatibilidad con las implementaciones de la interfaz de usuario de material design.
com.android.support: recyclerview	Proporciona compatibilidad con el widget RecyclerView, una vista para mostrar conjuntos de datos grandes proporcionando una cantidad limitada de elementos de datos.

<code>com.android.support: design</code>	Agrega compatibilidad con varios componentes y patrones de material design para referencia de los desarrolladores de apps, como los paneles laterales de navegación, los botones de acción flotantes (<i>FAB</i>), las snackbars y las pestañas.
--	--

6. Desarrollo del Proyecto

El proyecto está compuesto por cuatro módulos, los cuales se presentan en la Figura 2.

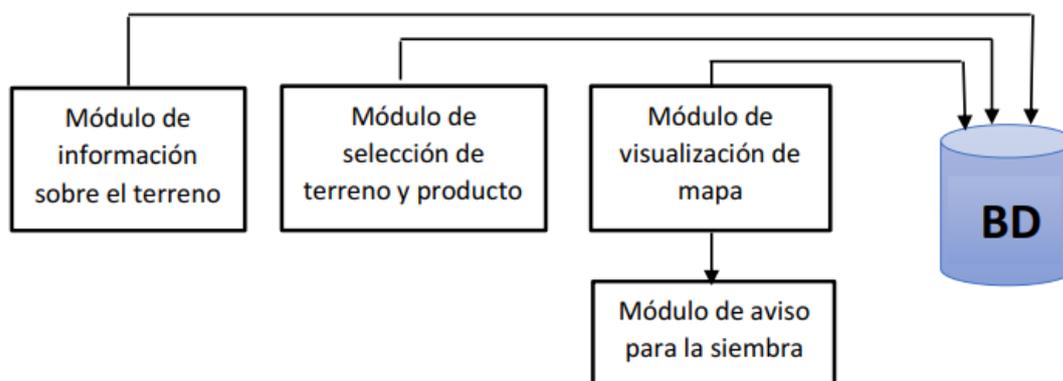


Figura 2. Módulos Componentes del Sistema

Durante el desarrollo fue necesario crear una base de datos en la cual se pudiera almacenar la información del Terreno (Información particular del terreno) y producto (Información sobre las dimensiones requeridas por los productos a sembrar) dicha base de datos se realizó en el gestor de base de datos SQLite y el diagrama Entidad Relación se presenta a continuación en la Figura 3.

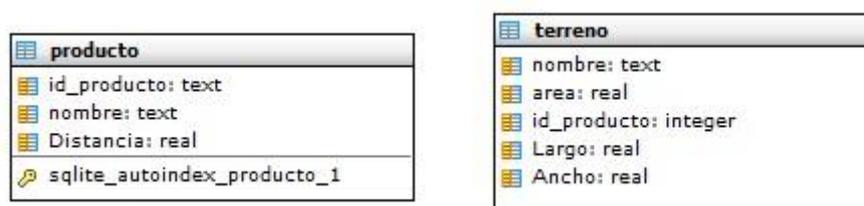


Figura 3. Diagrama Entidad Relación de Proyecto

Para comprender mejor la información del diagrama mostrado en la Figura 3 se presenta un diccionario de datos con las tablas pertenecientes a la base de datos del proyecto, esta información se muestra en las Tablas 3, 4 y 5.

7.1 Tablas

Tabla 3. Tablas que componen la Base de Datos utilizada en el proyecto.

NOMBRE	DESCRIPCIÓN
Terreno	Información sobre dimensiones del terreno
Producto	Información de dimensión requerida por cada producto a sembrar

Tabla 4. Terreno

COLUMNA	TIPO	DESCRIPCIÓN	RESTRICCIONES
id_terreno	PK	Identificador del terreno	Solo puede contener números. No Null
Nombre		nombre del terreno	No Null
Área		área del terreno	No Null
id_producto	FK	Identificador del producto	Solo puede contener números. No Null

Tabla 5. Producto

COLUMNA	TIPO	DESCRIPCIÓN	RESTRICCIONES
id_producto	PK	Identifica el producto	Solo puede contener números. No Null
Nombre		Nombre del producto	No Null
Distancia		Distancia requerida por producto	No Null

Los casos de uso así como los diagramas de secuencia permiten comprender la interacción de nuestro usuario con la aplicación desarrollada. En la Figura 4 se muestra el Caso de uso Agregar Terreno en el cual se puede observar la forma en la que el usuario final logra trabajar con la sección de agregar terreno que se encuentra en la aplicación móvil desarrollada. Así mismo en la Figura 5 y 6 se

presenta el diagrama de secuencia que muestra la interacción de los elementos dentro de la aplicación.

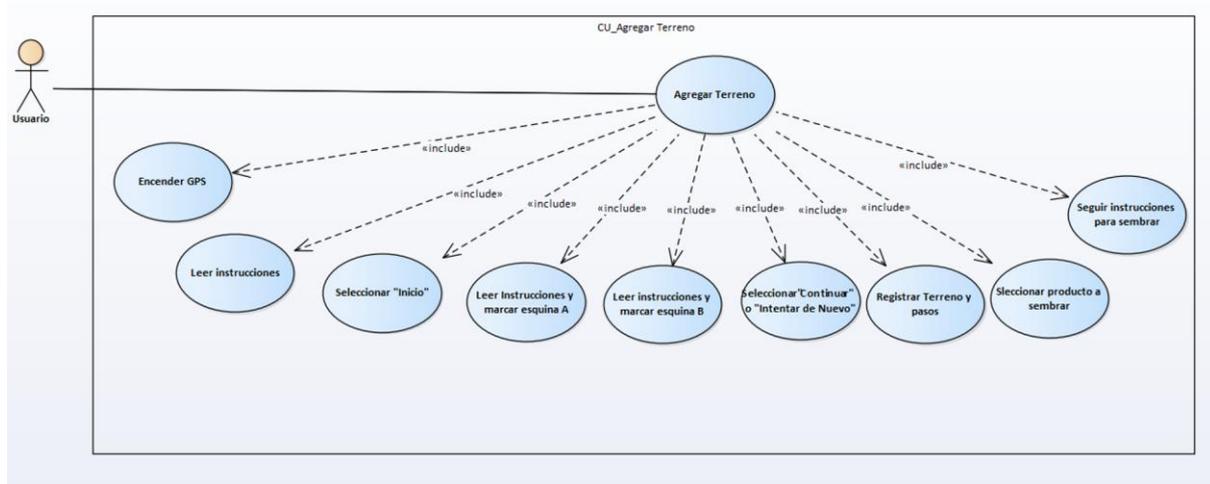


Figura 4. Caso de Uso: Agregar Terreno.

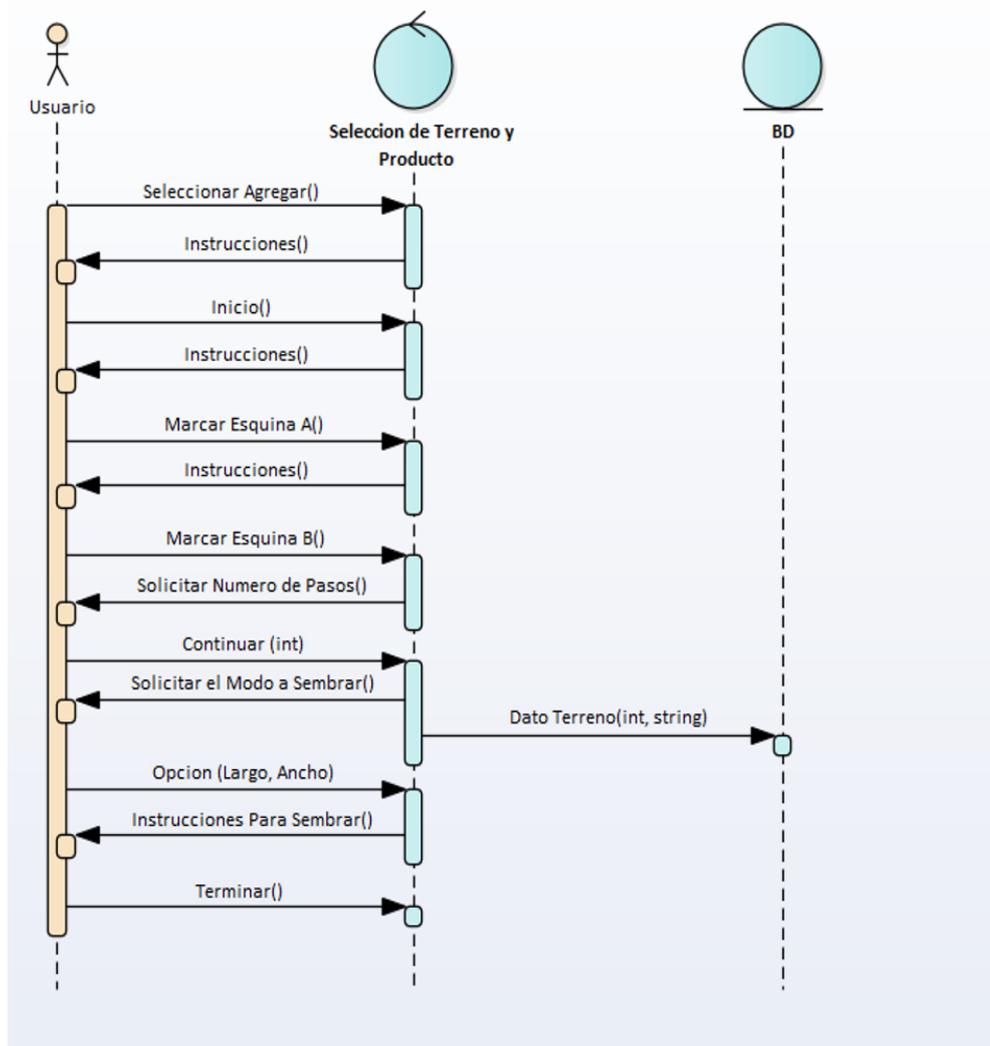


Figura 5. Diagrama de Secuencia en Caso de Éxito.

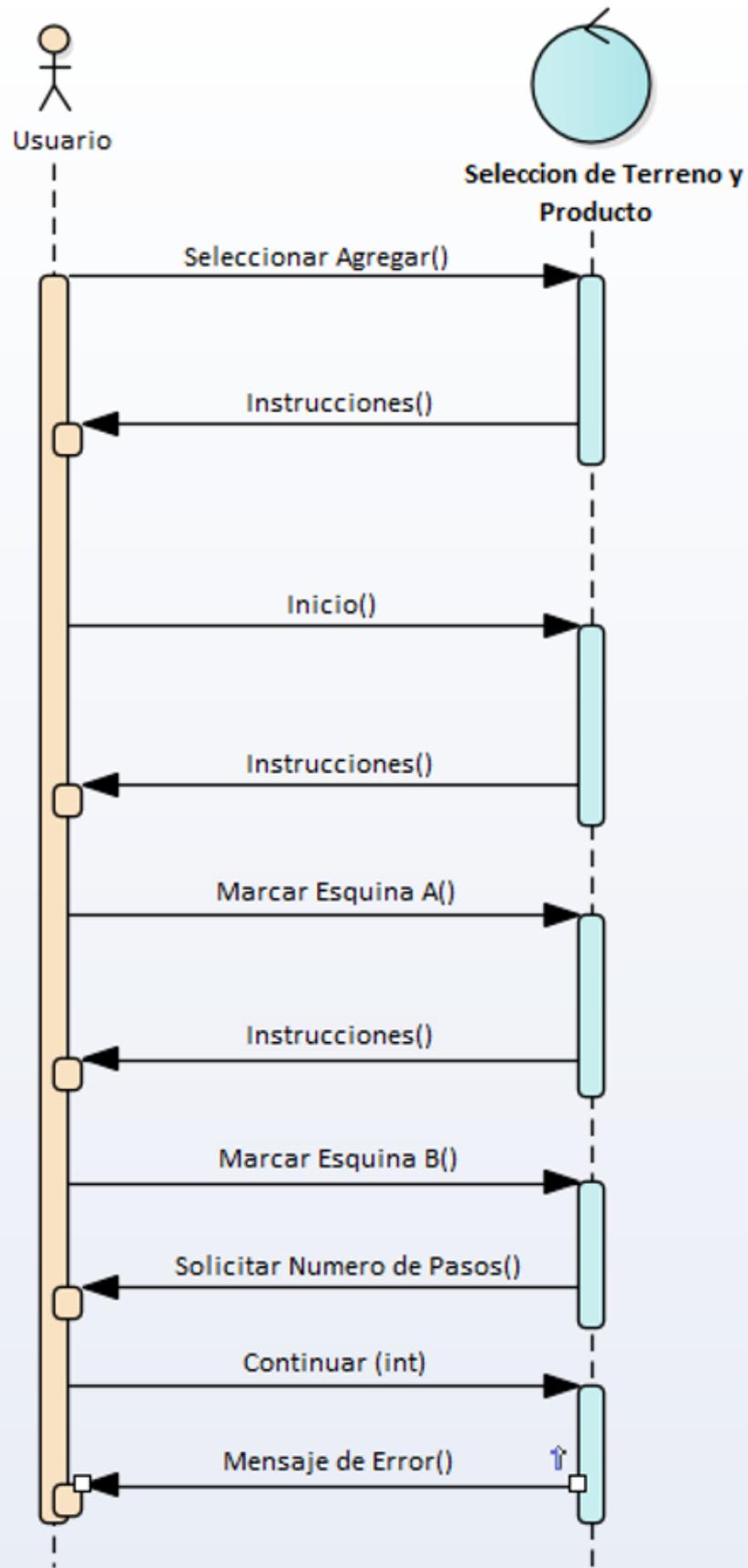


Figura 6. Diagrama de Secuencia en Caso de Fracaso.

Fue necesario realizar un diseño de la interfaz del proyecto a realizar para poder determinar cuáles serían las posibles complicaciones que tendría el usuario al interactuar con la aplicación. En las Figuras de la 7 a la 21 presentadas a continuación se muestran las interfaces diseñadas y una breve descripción de ellas.

 <p>BIENVENIDO Seleccione una opción</p> <p>AGREGAR</p>	 <p>NUEVO TERRENO</p> <p>Posiciónate en una esquina mirando hacia el centro del terreno y presiona el boton iniciar.</p> <p>INICIAR</p>	 <p>NUEVO TERRENO</p> <p>Ahora camina a la izquierda o derecha en línea recta hacia otra esquina del terreno. Cuenta CUANTOS PASOS DAS. Camina de lo más normal, entre mas natural sea, mejor será mi precisión. Al llegar presiona el botón marcar</p> <p>MARCAR ESQUINA A</p>
 <p>NUEVO TERRENO</p> <p>Repite el mismo procedimiento ahora hacia la siguiente esquina, de forma que hayamos formado una esquina de rectángulo. Recuerda llevar el conteo de tus pasos</p> <p>MARCAR ESQUINA B</p>	 <p>NUEVO TERRENO</p> <p>AHORA DEBERÁS INGRESAR LOS PASOS QUE DISTE.</p> <p>¿Te equivocaste? Pulsa el botón "De nuevo" para realizar las mediciones otra vez</p> <p>CONTINUAR</p> <p>¡DE NUEVO!</p>	 <p>sonia</p> <p>PASOS 40</p> <p>PLANTA A SEMBRAR Aguacate</p> <p>CONTINUAR</p>
<p>Figura 7. Inicio de la aplicación móvil.</p>	<p>Figura 8. Punto de partida para datos del terreno.</p>	<p>Figura 9. Segundo punto considerado para obtener un lado.</p>
<p>Figura 10. Último punto necesario para calcular el área del terreno.</p>	<p>Figura 11. Permite volver a medir o continuar con el proceso de siembra.</p>	<p>Figura 12. Al continuar se ingresan el nombre, los pasos y el producto a sembrar (aguacate).</p>



Figura 13. Registro de terreno con nombre, pasos dados y producto a sembrar (Maíz).



Figura 14. Registro de terreno con nombre, pasos dados y producto a sembrar (Naranja).



Figura 15. Detalles del terreno, dimensiones, área, etc.



Figura 16. Seleccionar la forma de sembrar.



Figura 17. Indicaciones de siembra.



Figura 18. Botón Cargar para visualizar los terrenos guardados.

<p style="text-align: center;">SELECCIONA TU TERRENO</p> <div style="background-color: #4CAF50; color: white; padding: 5px; display: flex; justify-content: space-between; align-items: center;"> sonia </div>	<div style="text-align: center;"> </div> <p>PASOS 40</p> <hr style="width: 50%; margin: 0 auto;"/> <p>PLANTA A CEMBRAD Naranja ▼</p> <div style="background-color: #4CAF50; color: white; text-align: center; padding: 10px; margin-top: 10px;">CONTINUAR</div>	<div style="text-align: center;"> <p>centro del terreno</p> <p>Continua hacia el frente 100 + 1/4 pasos</p> <p>Repite por 2 gira a la derecha y camina 100 + 1/4 pasos</p> <p>Repite desde el paso uno al tres por 2 veces mas, hasta haber completado el area del terreno</p> <div style="background-color: #4CAF50; color: white; text-align: center; padding: 5px; margin-top: 5px;">TERMINAR</div> </div>
<p>Figura 19. Lista de terrenos guardados.</p>	<p>Figura 20. Seleccionar producto a sembrar.</p>	<p>Figura 21. Indicaciones de cómo realizar la siembra.</p>

7. Resultados

Una vez concluida la aplicación se realizaron una serie de pruebas en el campo de futbol de la UAM-Azcapotzalco, se escogió ese lugar porque cumplía con los requisitos del terreno para poder sembrar, es decir era plano y las áreas seleccionadas para sembrar son cuadradas, durante la realización de las pruebas se obtuvieron los resultados mostrados en la Tablas 6.

Tabla 6. Resultados de Pruebas Realizadas					
No. DE PRUEBA	LARGO	ANCHO	ÁREA	PRODUCTO	PLANTAS (TOTAL)
01	67.1	63.9	4284.332	Maíz	5712
02	113.7	67.4	7667.844	Maíz	10223
03	49.5	47.1	2333.4019	Maíz	3111
04	114.5	66.8	7649.9834	Aguacate	1274
05	113.9	66.4	7567.881	Aguacate	1261
06	67.3	51.8	3485.802	Aguacate	580
07	66.6	46.0	3063.3682	Naranja	612
08	69.9	41.8	2925.5322	Naranja	585
09	59.4	33.4	1985	Maíz	2647
10	114.4	36.6	4193.469	Maíz	5591

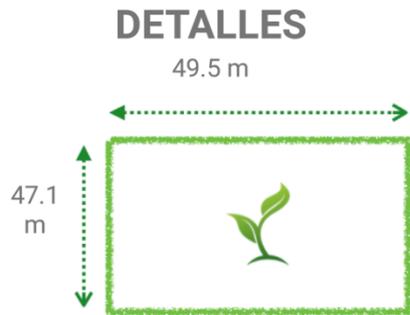
Las capturas de pantalla correspondientes a las pruebas realizadas se presentan en las Figuras de la 22 a la 31.



Figura 22. Resultados de Prueba01



Figura 23. Resultados de Prueba02



ÁREA DEL TERRENO: 2333.4019 m²

#PLANTAS A SEMBRAR: 3111

DISTANCIA ENTRE PLANTAS 75.0cm

¿CÓMO VA A SEMBRAR?



Figura 24. Resultados de Prueba03



ÁREA DEL TERRENO: 7649.9834 m²

#PLANTAS A SEMBRAR: 1274

DISTANCIA ENTRE PLANTAS 600.0cm

¿CÓMO VA A SEMBRAR?



Figura 25. Resultados de Prueba04



Figura 26. Resultados de Prueba05



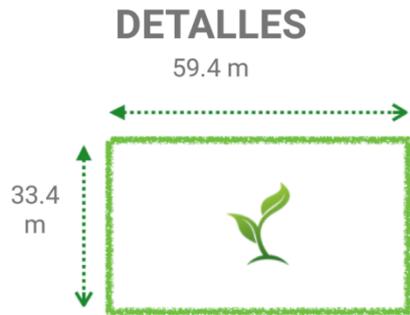
Figura 27. Resultados de Prueba06



Figura 28. Resultados de Prueba07



Figura 29. Resultados de Prueba08

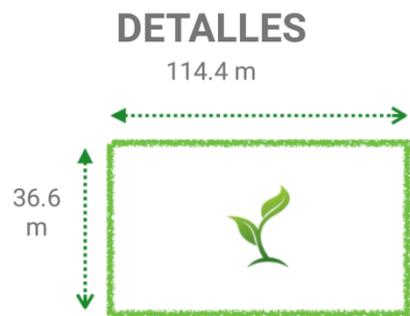


ÁREA DEL TERRENO:	1985.413 m2
#PLANTAS A SEMBRAR:	2647
DISTANCIA ENTRE PLANTAS	75.0cm

¿CÓMO VA A SEMBRAR?



Figura 30. Resultados de Prueba09



ÁREA DEL TERRENO:	4193.469 m2
#PLANTAS A SEMBRAR:	5591
DISTANCIA ENTRE PLANTAS	75.0cm

¿CÓMO VA A SEMBRAR?



Figura 31. Resultados de Prueba10

8. Conclusiones

Una vez finalizado el proyecto y analizado los resultados de las pruebas se puede concluir que la Aplicación es una herramienta adecuada que facilita a las personas dedicadas a la siembra la colocación de las semillas de los producto gracias al cálculo del área del terreno y debido a que cuenta con una interfaz sencilla e intuitiva se convierte en una herramienta muy importante para los agricultores.

Además la realización de este proyecto me permitió ampliar mis conocimientos de programación y otras tecnologías al tener que trabajar con entornos de desarrollo que desconocía.

Para finalizar es importante decir que los objetivos planteados por el proyecto se cumplieron de forma satisfactoria y por ellos se concluye que la aplicación se realizó con éxito y que el proyecto está concluido.

9. Bibliografía.

[1] J. J. García Rodríguez, M. A. Ávila Perches, J. D. De la Torre Vizcaino, C. Herrera Corredor, "Diferentes Patrones de Siembra en la Producción de Semilla de Híbrido de Maíz H-135", vol.14,no. 1, pp 45-46, 2014.

[2] J. R. Castillo Velazquez, "Transmisión y registro de las coordenadas geográficas de un dispositivo móvil", Proyecto Terminal, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, México, 2012.

[3] A. E. Villarruel Barajas, "Asignación óptima para el uso de laboratorios y talleres de la División de Ciencias Basicas e Ingenieria", Proyecto Terminal, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, México, 2012.

[4] L. A. Carrasco Reyes, "Sistema de Información Web para la Geolocalización de Unidades Aers", Proyecto Terminal. División de Ciencias Basicas e Ingenieria, Universidad Autonoma Metropolitana Azcapotzalco, México, 2015.

[5] P. Donelan, "Cultivo de Semillas", Ecology Action, vol. 3, no. 13. pp. 43-44, 2009.

[6] G. DPI, "Mide Mapas - by global DPI en App store," App Store, 2016. Available: <https://itunes.apple.com/mx/app/mide-mapas-by-global-dpi/id417622293?mt=8>. Accessed: Mar. 1, 2017.

[7] T. S. de CV, "Syngenta Soluciones en App store," App Store, 2016. Available: <https://itunes.apple.com/mx/app/syngenta-soluciones/id1020856540?mt=8>. Accessed: Mar. 1, 2017.

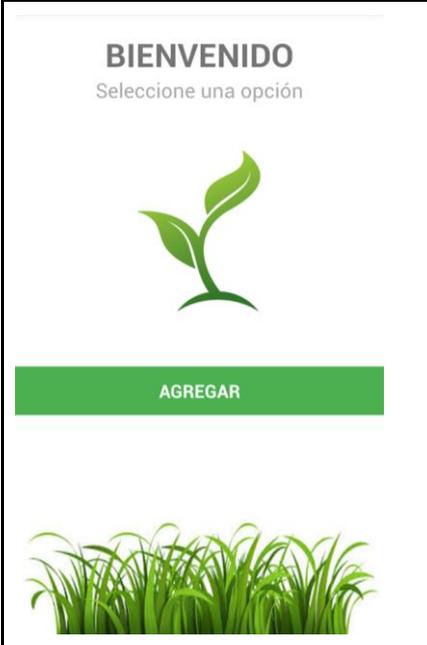
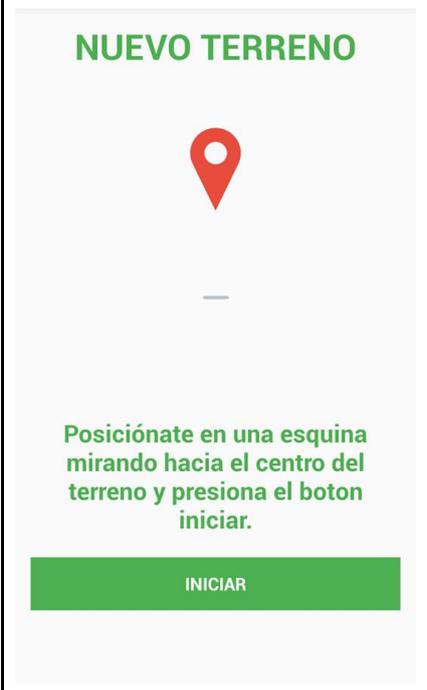
[8] Gustavo N. Ferraris, "Comportamiento de diferentes genotipos de maíz frente a variaciones en fechas de siembra,densidad y fertilización nitrogenada", Instituto Nacional de Tecnología Agropecuaria, Argentina, 2016.

[9] "Método y densidad de siembra del cultivo de Maíz", AgroSíntesis, 2016. [Online]. Available: www.agrosintesis.com/. Accessed: 15- Junio- 2017.

[10] "Distancia de siembra en el cultivo de cítricos", Instituto Nacional Autónomo de Investigaciones Agropecuarias, 2014. [Online]. Available: www.iniap.gob.ec/nsite/. Accessed: 15- Junio- 2017.

[11] "INIAP evalúa distancia de siembra en el cultivo de aguacate con buenos resultados", Instituto Nacional Autónomo de Investigaciones Agropecuarias, 2013. [Online]. Available: www.iniap.gob.ec/nsite/. Accessed: 15- Junio- 2017.

10. Entregables

DISEÑO DE MÓDULO I: MÓDULO DE INFORMACIÓN DEL TERRENO	
CAPTURA	DESCRIPCIÓN
	<p>Pantalla I.</p> <p>Presenta mensaje de bienvenida al usuario y muestra el botón agregar para que el usuario pueda registrar su primer terreno en el que podrá sembrar su producto.</p>
	<p>Pantalla II.</p> <p>En esta pantalla el usuario deberá leer las instrucciones y mediante el botón Iniciar el usuario podrá comenzar con el registro de los puntos de su terreno.</p>

<p>NUEVO TERRENO</p>  <p>—</p> <p>Ahora camina a la izquierda o derecha en línea recta hacia otra esquina del terreno. Cuenta CUANTOS PASOS DAS. Camina de lo más normal, entre mas natural sea, mejor será mi precisión. Al llegar presiona el botón marcar</p> <p>MARCAR ESQUINA A</p>	<p>Pantalla III.</p> <p>En esta pantalla el usuario podrá marcar el segundo punto de su terreno, para ello debe seguir las indicaciones mostradas en la pantalla y oprimir el botón MARCAR ESQUINA A, de esta forma habrá obtenido la distancia que tiene un lado de su terreno cuadrilátero.</p>
<p>NUEVO TERRENO</p>  <p>—</p> <p>Repite el mismo procedimiento ahora hacia la siguiente esquina, de forma que hayamos formado una esquina de rectángulo. Recuerda llevar el conteo de tus pasos</p> <p>MARCAR ESQUINA B</p>	<p>Pantalla IV.</p> <p>En esta pantalla el usuario podrá marcar último punto de su terreno, para ello debe seguir las indicaciones mostradas en la pantalla y oprimir el botón MARCAR ESQUINA B. Al presionar el botón se habrá obtenido el segundo lado del terreno cuadrilátero y se podrá hacer el cálculo del área.</p>

<p>NUEVO TERRENO</p>  <p>—</p> <p>AHORA DEBERÁS INGRESAR LOS PASOS QUE DISTE.</p> <p>¿Te equivocaste? Pulsa el botón "De nuevo" para realizar las mediciones otra vez</p> <p>CONTINUAR</p> <p>¡DE NUEVO!</p>	<p>Pantalla V. En esta pantalla el usuario podrá volver a iniciar su medición o podrá continuar con el proceso de registro de terreno perteneciente al módulo de Información del Terreno.</p>
<p>BIENVENIDO Seleccione una opción</p>  <p>AGREGAR CARGAR</p> 	<p>Pantalla VI. En esta pantalla el usuario podrá seleccionar entre agregar un nuevo terreno o cargar un terreno que haya guardado previamente.</p>

SELECCIONA TU TERRENO

sonia



Pantalla VII.

En esta pantalla el usuario podrá encontrar la lista de los terrenos que hayan sido guardados previamente, y al tocar el nombre del terreno serán enviados a la pantalla X.

CÓDIGO DE MÓDULO I: MÓDULO DE INFORMACIÓN DEL TERRENO

campo.java

```
package com.example.sonia.azadon;

import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteException;
import android.location.Location;
import android.location.LocationManager;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;

import java.util.ArrayList;

public class campo extends AppCompatActivity {

    Button btn_agregar;
    Button btn_cargar;
    public static DatabaseHelper DB;
    public static Boolean cargar = false;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_campo);

        btn_agregar = (Button) findViewById(R.id.btn_agregar);
        btn_cargar = (Button) findViewById(R.id.btn_cargar);

        //Abriendo la base de datos

        DB = new DatabaseHelper(this);

        try {
            DB.copiarBasePrimeraVez();
        } catch (SQLiteException e) {
            Log.e("@Sonia agregar.java", "Error al crear la base");
        }

        try {
            DB.abrirBase();
        } catch (SQLiteException esql) {
            Log.e("@Sonia agregar.java", "Error al abrir la base de
datos");
        }

        //Verificar si hay datos

        btn_agregar.setOnClickListener(new View.OnClickListener() {
```

```

        @Override
        public void onClick(View v) {
            Intent i = new Intent(v.getContext(), agregar.class);
            startActivity(i);
        }
    });

    if (hayTerrenos()) {
        btn_cargar.setVisibility(View.VISIBLE);
        btn_cargar.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i = new Intent(v.getContext(),
terreno_guardado.class);
                startActivity(i);
            }
        });
    } else {
        btn_cargar.setVisibility(View.GONE);
    }
}

public Boolean hayTerrenos () {
    Cursor c = DB.hacer_consulta("SELECT * FROM terreno ORDER BY
nombre;");

    if (c.moveToFirst()) return true;
    else return false;
}
}

```

CÓDIGO DE MÓDULO I: MÓDULO DE INFORMACIÓN DEL TERRENO

agregar.java

```
package com.example.sonia.azadon;

import android.Manifest;
import android.app.Dialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.database.Cursor;
import android.database.sqlite.SQLiteException;
import android.location.Criteria;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Build;
import android.provider.Settings;
import android.support.design.widget.Snackbar;
import android.support.v4.app.ActivityCompat;
import android.support.v4.content.ContextCompat;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import org.w3c.dom.Text;

import java.io.IOException;
import java.util.ArrayList;

import static com.example.sonia.azadon.campo.DB;

public class agregar extends AppCompatActivity implements LocationListener
{
    private TextView tv_instrucciones;
    private TextView tv_wrong;

    private Button btn_agregar_ab;
    private Button btn_agregar_bc;
    private Button btn_iniciar;
    private Button btn_reset;
    private Button btn_continuar;

    public static float base, altura, aux, distanciaTotal, area;
    private double latitud, longitud;
```

```

private double latA, latB, latC, longA, longB, longC;

public LocationManager gps;
public Location ubicacion;

/*Esta clase está asociada a la pantalla "Agregar terreno", la cual
permite hacer nuestra caminata para poder hacer las mediciones del
terreno
asi mismo, calcula la distancia para después poder sacar nuestro
"paso promedio"
*/

@Override
protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);
    setContentView(R.layout.agregar);

    /*Instanciar elementos de la interfaz gráfica
    botones, Cuadros de texto*/

    btn_agregar_ab = (Button) findViewById(R.id.btn_agregar_ab);
    btn_agregar_bc = (Button) findViewById(R.id.btn_agregar_bc);
    btn_iniciar = (Button) findViewById(R.id.btn_iniciar);
    btn_reset = (Button) findViewById(R.id.btn_reset);
    btn_continuar = (Button) findViewById(R.id.btn_continuar);

    tv_instrucciones = (TextView) findViewById(R.id.tv_instrucciones);
    tv_wrong = (TextView) findViewById(R.id.tv_wrong);

    btn_agregar_ab.setVisibility(View.GONE);
    btn_agregar_bc.setVisibility(View.GONE);
    btn_reset.setVisibility(View.GONE);
    tv_wrong.setVisibility(View.GONE);
    btn_continuar.setVisibility(View.GONE);

    //Escuchador de evento, en este caso al presionar
    //un botón (onClick)

    btn_iniciar.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            tv_wrong.setVisibility(View.GONE);

            try {
                if (IniciarServicio(v)) {

                    mostrarPosicionActual();
                    latA = latitud;
                    longA = longitud;

                    btn_agregar_ab.setVisibility(View.VISIBLE);
                    btn_iniciar.setVisibility(View.GONE);
                    tv_instrucciones.setText("Ahora camina a la
izquierda o derecha en linea recta hacia otra esquina del terreno. Cuenta
CUANTOS PASOS DAS." +

                                " Camina de lo más normal, entre mas
natural sea, mejor será mi precisión. Al llegar presiona el botón
marcar");
                }
            }
        }
    });

```

```

        } else {
            Log.e("@Sonia", "Error con los permisos");
            Toast.makeText(getApplicationContext(), "Error con
los permisos", Toast.LENGTH_SHORT);
        }
    } catch (InterruptedException e) {
        Log.d("@Sonia", "Error con el gps");
    }
}
});

btn_agregar_ab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        try {
            if (IniciarServicio(v)) {
                mostrarPosicionActual();
                tv_instrucciones.setText("Repite el mismo
procedimiento ahora hacia la siguiente esquina, de forma que hayamos
formado una esquina de rectángulo. Recuerda llevar el conteo de tus
pasos");

                latB = latitud;
                longB = longitud;
                base = (float) distancia(latA, longA, latB,
longB);

                btn_agregar_bc.setVisibility(View.VISIBLE);
                btn_iniciar.setVisibility(View.GONE);
                btn_agregar_ab.setVisibility(View.GONE);

            } else {
                Log.e("@Sonia", "Error con los permisos");
                Toast.makeText(getApplicationContext(), "Error con
los permisos", Toast.LENGTH_SHORT);
            }
        } catch (InterruptedException e) {
            Log.d("@Sonia", "Error con el gps");
        }
    }
});

btn_agregar_bc.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        try {
            if (IniciarServicio(v)) {
                mostrarPosicionActual();

                btn_agregar_ab.setVisibility(View.GONE);
                btn_agregar_bc.setVisibility(View.GONE);
                btn_reset.setVisibility(View.VISIBLE);
                tv_wrong.setVisibility(View.VISIBLE);
                btn_continuar.setVisibility(View.VISIBLE);

                tv_instrucciones.setText("Ahora deberás ingresar
los pasos que diste.");

```

```

        tv_instrucciones.setAllCaps(true);

        latC = latitud;
        longC = longitud;
        altura = (float) distancia(latB, longB, latC,
longC);

        } else {
            Log.e("@Sonia", "Error con los permisos");
            Toast.makeText(getApplicationContext(), "Error con
los permisos", Toast.LENGTH_SHORT);
        }
    } catch (InterruptedException e) {
        Log.d("@Sonia", "Error con el gps");
    }
}

});

btn_reset.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        btn_agregar_bc.setVisibility(View.GONE);
        btn_reset.setVisibility(View.GONE);
        btn_continuar.setVisibility(View.GONE);
        tv_wrong.setVisibility(View.GONE);
        btn_iniciar.setVisibility(View.VISIBLE);
        tv_instrucciones.setText("Posiciónate en una esquina
mirando hacia el centro del terreno y presiona el boton iniciar.");
    }
});

btn_continuar.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent i = new Intent(v.getContext(), pasos.class);

        //Evalua si la base es mas grande que la altura
        //e invierte valores de ser asi
        if (base < altura) {
            aux = base;
            base = altura;
            altura = aux;
        }

        area = Math.round(base * altura);
        distanciaTotal = Math.round(base + altura);
        area *= 100;
        distanciaTotal *= 100;
        base *= 100;
        altura *= 100;
        Log.i("@Sonia Agregar", "Base: " + base + " Altura: " +
altura + " DistanciaTotal: " + distanciaTotal + "Area" + area);

        if (base > 0 && altura > 0)
            startActivity(i);
        else {
            AlertDialog.Builder builder = new

```

```

AlertDialog.Builder(agregar.this);
    builder.setMessage("Debes seguir las instrucciones y
moverte de lugar, para que yo pueda medir el terreno.");
    builder.setCancelable(false);
    builder.setPositiveButton("OK", new
DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int
which) {
        dialog.cancel();
    }
});

AlertDialog alerta = builder.create();
alerta.show();
}

});
}

/*
Inicia el servicio del GPS en el dispositivo, debe de esperarse unos
segundos para
que se haga correctamente
*/
public Boolean IniciarServicio(View v) throws InterruptedException {

    String proveedor;
    gps = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);

    Snackbar.make(v, "Ubicación detectada",
Snackbar.LENGTH_LONG).show();

    Criteria c = new Criteria();
    c.setAccuracy(Criteria.ACCURACY_FINE);

    proveedor = gps.getBestProvider(c, true);
    Toast.makeText(this, "Proveedor: " + proveedor,
Toast.LENGTH_SHORT);
    Log.d("@Sonia", "Proveedor: " + proveedor);

    if (Build.VERSION.SDK_INT >= 23) {
        if (ContextCompat.checkSelfPermission(this,
android.Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED
        || ContextCompat.checkSelfPermission(this,
android.Manifest.permission.ACCESS_COARSE_LOCATION) ==
PackageManager.PERMISSION_GRANTED) {

            if (proveedor == null || proveedor.equals("")) {
                startActivity(new
Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
            } else {
                for (int j = 0; j < 10; j++) {
                    gps.requestLocationUpdates(proveedor, 1000, 0,
this);

                    ubicacion = gps.getLastKnownLocation("gps");

```

```

    }
}

return true;

} else {

    final int MY_PERMISSION_ACCESS_COARSE_LOCATION = 11;
    final int MY_PERMISSION_ACCESS_FINE_LOCATION = 12;

    // The ACCESS_COARSE_LOCATION is denied, then I request it
and manage the result in
    // onRequestPermissionsResult() using the constant
MY_PERMISSION_ACCESS_FINE_LOCATION
    if (ContextCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this,
            new
String[]{Manifest.permission.ACCESS_COARSE_LOCATION},
            MY_PERMISSION_ACCESS_COARSE_LOCATION);
    }
    // The ACCESS_FINE_LOCATION is denied, then I request it
and manage the result in
    // onRequestPermissionsResult() using the constant
MY_PERMISSION_ACCESS_FINE_LOCATION
    if (ContextCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this,
            new
String[]{Manifest.permission.ACCESS_FINE_LOCATION},
            MY_PERMISSION_ACCESS_FINE_LOCATION);
    }
}

return false;

} else {
    // At least one provider activated. Get the coordinates
gps.requestLocationUpdates(proveedor, 100, 1, this);
ubicacion = gps.getLastKnownLocation(proveedor);

return true;
}

}

//Obtiene nuestra latitud y longitud cada vez que marcamos una esquina
public void mostrarPosicionActual() {

    if (ubicacion == null) {
        Log.e("@Sonia", "Location -> NULL");
    } else {
        longitud = ubicacion.getLongitude();
        latitud = ubicacion.getLatitude();
    }
}

```

```

    }

    public void pararServicio() {
        gps.removeUpdates(this);
    }

    /*Mediante la fórmula de Haversine, nos permite calcular entre dos
    coordenadas la distancia
    entre si
    */

    public double distancia(double lat1, double long1, double lat2, double
    long2) {

        //Utilizando la formula de Haversine

        final double radiotierra = 6371000.00;
        double DeltaLat, DeltaLong, distancia, angulo;

        DeltaLat = lat2 - lat1;
        DeltaLong = long2 - long1;

        angulo = 2 * Math.asin(Math.sqrt(Math.pow(Math.sin(toRad(DeltaLat
        / 2)), 2) + Math.cos(toRad(lat1)) * Math.cos(toRad(lat2)) *
        Math.pow(Math.sin(toRad(DeltaLong / 2)), 2)));

        return angulo * radiotierra;

    }

    //Función convertidora a Radianes
    public double toRad(double grados) {
        return grados * Math.PI / 180;
    }

    //Funciones de control del GPS

    @Override
    public void onLocationChanged(Location location) {

    }

    @Override
    public void onStatusChanged(String provider, int status, Bundle
    extras) {

    }

    @Override
    public void onProviderEnabled(String provider) {

    }

    @Override
    public void onProviderDisabled(String provider) {

    }

}

```

CÓDIGO DE MÓDULO I: MÓDULO DE INFORMACIÓN DEL TERRENO

Terreno_guardado.java

```
package com.example.sonia.azadon;

import android.database.Cursor;
import android.database.sqlite.SQLiteException;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.util.Log;

import static com.example.sonia.azadon.campo.cargar;

import java.util.ArrayList;
import java.util.List;

import static com.example.sonia.azadon.campo.DB;

public class terreno_guardado extends AppCompatActivity {

    /*Esta clase permite cargar un terreno previamente guardado
    en la DB, nos deja reingresar los pasos por si es otra persona quien
    quiere sembrar en el mismo terreno, además de volver a elegir la
    planta
    */

    RecyclerView reciclador;
    public List<terreno> listaTerrenos;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        cargar = true;
        setContentView(R.layout.activity_terreno_guardado);

        reciclador = (RecyclerView) findViewById(R.id.reciclador);
        LinearLayoutManager lymanager = new
        LinearLayoutManager(getApplicationContext());
        reciclador.setLayoutManager(lymanager);

        listaTerrenos = new ArrayList<>();

        //Abriendo la base de datos

        DB = new DatabaseHelper(this);

        try {
            DB.copiarBasePrimeraVez();
        } catch (SQLiteException e) {
            Log.e("@Sonia agregar.java", "Error al crear la base");
        }
    }
}
```

```

        try {
            DB.abrirBase();
        } catch (SQLException esql) {
            Log.e("@Sonia agregar.java", "Error al abrir la base de
datos");
        }

        cargarTerrenos();

        adaptador_terreno adapter = new adaptador_terreno(listaTerrenos);
        reciclador.setAdapter(adapter);

    }

    public void cargarTerrenos() {

        Cursor c = DB.hacer_consulta("SELECT * FROM terreno;");

        if (c.moveToFirst()) {
            do {
                terreno t = new terreno();
                t.setNombre(c.getString(0));
                t.setArea(c.getFloat(1));
                t.setBase(c.getFloat(2));
                t.setAltura(c.getFloat(3));
                listaTerrenos.add(t);
                Log.d("@Sonia TERRENOS", "Nombre: " + t.getNombre() + "
Area: " + t.getArea());
            } while (c.moveToNext());
        }

    }

}

```

DISEÑO DE MÓDULO II: MÓDULO DE SELECCIÓN DE TERRENO Y PRODUCTO

CAPTURA	DESCRIPCIÓN
	<p>Pantalla VIII.</p> <p>En esta pantalla el usuario podrá registrar su terreno el cual fue medido en el módulo de Información del terreno, el usuario podrá nombrar su terreno y deberá ingresar el número total de pasos que dió para poder realizar el cálculo de distancia que tienen sus propios pasos ya que esta información se utilizará posteriormente, además seleccionará uno de los 3 productos que se ofrecen en la aplicación, en este caso se presenta el producto Aguacate.</p>
	<p>Pantalla IX.</p> <p>En esta pantalla el usuario podrá registrar su terreno el cual fue medido en el módulo de Información del terreno, el usuario podrá nombrar su terreno y deberá ingresar el número total de pasos que dió para poder realizar el cálculo de distancia que tienen sus propios pasos ya que esta información se utilizará posteriormente, además seleccionará uno de los 3 productos que se ofrecen en la aplicación, en este caso se presenta el producto Maíz.</p>



Pantalla X.

En esta pantalla el usuario podrá registrar su terreno el cual fue medido en el módulo de Información del terreno, el usuario podrá nombrar su terreno y deberá ingresar el número total de pasos que dió para poder realizar el cálculo de distancia que tienen sus propios pasos ya que esta información se utilizará posteriormente, además seleccionará uno de los 3 productos que se ofrecen en la aplicación, en este caso se presenta el producto Naranja.

CÓDIGO DE MÓDULO II: MÓDULO DE SELECCIÓN DE TERRENO Y PRODUCTO

pasos.java

```
package com.example.sonia.azadon;

import android.content.ContentValues;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteException;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import java.util.ArrayList;

import static com.example.sonia.azadon.DatabaseHelper.DBcampo;
import static com.example.sonia.azadon.agregar.altura;
import static com.example.sonia.azadon.agregar.area;
import static com.example.sonia.azadon.agregar.base;
import static com.example.sonia.azadon.agregar.distanciaTotal;
import static com.example.sonia.azadon.campo.DB;
import static com.example.sonia.azadon.campo.cargar;

public class pasos extends AppCompatActivity {

    private Button btn_continue_to_sembrar;
    private EditText ed_pasos;
    private EditText ed_nombre_terreno;
    private Spinner spinner_semillas;
    private int idDeProducto;
    public static float paso_promedio;
    public static float distanciaEntrePlantas;

    ArrayList<semilla> semillas = new ArrayList<>();
    public static String[] semillasEnSpinner = null;

    @Override
    protected void onCreate(final Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_pasos);

        btn_continue_to_sembrar = (Button)
        findViewById(R.id.btn_continuar_to_sembrar);
        ed_pasos = (EditText) findViewById(R.id.ed_pasos);
        ed_nombre_terreno = (EditText)
        findViewById(R.id.ed_nombre_terreno);
        spinner_semillas = (Spinner) findViewById(R.id.spinner_semillas);
```

```

//Llenar el Spinner

getSemillasSpinner();

    ArrayAdapter<String> adaptador_spinner = new
ArrayAdapter<String>(getApplicationContext(),
android.R.layout.simple_spinner_item, semillasEnSpinner);

adaptador_spinner.setDropDownViewResource(android.R.layout.simple_spinner_d
ropdown_item);
    spinner_semillas.setAdapter(adaptador_spinner);

    spinner_semillas.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view,
int position, long id) {
            ((TextView) parent.getChildAt(0)).setTextSize(20);
            ((TextView)
parent.getChildAt(0)).setTextColor(getResources().getColor(R.color.texto_ne
gro));

            distanciaEntrePlantas =
semillas.get(position).getDistancia();
            idDeProducto = semillas.get(position).getId_producto();
        }

        @Override
        public void onNothingSelected(AdapterView<?> parent) {
            ((TextView) parent.getChildAt(0)).setTextSize(20);
            ((TextView)
parent.getChildAt(0)).setTextColor(getResources().getColor(R.color.texto_ne
gro));
        }
    });

    if (!cargar) {

        ed_nombre_terreno.setVisibility(View.VISIBLE);
        Log.d("@Sonia", "Se recibe altura: " + altura + " base: " +
base);

        DB = new DatabaseHelper(this);

        try {
            DB.copiarBasePrimeraVez();
        } catch (SQLException e) {
            Log.e("@Sonia agregar.java", "Error al crear la base");
        }

        try {
            DB.abrirBase();
        } catch (SQLException esql) {
            Log.e("@Sonia agregar.java", "Error al abrir la base de
datos");
        }

        btn_continue_to_sembrar.setOnClickListener(new
View.OnClickListener() {
            @Override

```

```

        public void onClick(View v) {
            Intent i = new Intent(v.getContext(),
mostrar_calculos.class);

            if (ed_pasos.getText().toString().equals("") &&
ed_nombre_terreno.getText().toString().equals("")) {
                Toast.makeText(v.getContext(), "LLENA TODOS LOS
CAMPOS", Toast.LENGTH_SHORT).show();
            } else {

                if (!ed_pasos.getText().toString().equals("")) {
                    if
((Integer.parseInt(ed_pasos.getText().toString()) > 0))
                    paso_promedio = (float) (distanciaTotal /
Integer.parseInt(ed_pasos.getText().toString()));
                    paso_promedio /= 10;
                    Log.d("@Sonia", "Dis: " + distanciaTotal + " "
+ Integer.parseInt(ed_pasos.getText().toString()) + " Div: " + (float)
(distanciaTotal / Integer.parseInt(ed_pasos.getText().toString())));
                } else {
                    Toast.makeText(pasos.this, "INGRESA LOS PASOS",
Toast.LENGTH_SHORT).show();
                }

                if
(!ed_nombre_terreno.getText().toString().equals("")) {
                    i.putExtra("nombre_terreno",
ed_nombre_terreno.getText().toString());
                } else {
                    Toast.makeText(v.getContext(), "INGRESA EL
NOMBRE DEL TERRENO", Toast.LENGTH_SHORT).show();
                }
            }

            if (!ed_pasos.getText().toString().equals("") &&
!ed_nombre_terreno.getText().toString().equals("")) {
                if
((Integer.parseInt(ed_pasos.getText().toString()) > 0)) {
                    guardarTerreno(ed_nombre_terreno.getText().toString(), area, base, altura);
                    startActivity(i);
                }
            }
        }
    });

} else {
    ed_nombre_terreno.setVisibility(View.GONE);
    cargar = false;

    btn_continue_to_sembrar.setOnClickListener(new
View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent i = new Intent(v.getContext(), sembrado.class);

            if (!ed_pasos.getText().toString().equals("")) {

```

```

        if
        ((Integer.parseInt(ed_pasos.getText().toString()) > 0)) {
            distanciaTotal = base + altura;
            paso_promedio = (float) (distanciaTotal /
Integer.parseInt(ed_pasos.getText().toString()));
            paso_promedio /= 10;
            Log.d("@Sonia", "Dis: " + distanciaTotal + " "
+ Integer.parseInt(ed_pasos.getText().toString()) + " Div: " + (float)
(distanciaTotal / Integer.parseInt(ed_pasos.getText().toString())));
            v.getContext().startActivity(i);
        }

        } else {
            Toast.makeText(pasos.this, "INGRESA LOS PASOS",
Toast.LENGTH_SHORT).show();
        }

    }

});

}

}

public String[] getSemillasSpinner() {

    Cursor c = DB.hacer_consulta("SELECT * FROM producto ORDER BY
nombre;");

    if (c.moveToFirst()) {
        do {
            semilla s = new semilla();
            s.setId_producto(c.getInt(0));
            s.setNombre(c.getString(1));
            s.setDistancia(c.getFloat(2));
            semillas.add(s);

            Log.d("@Sonia", "ID: " + s.getId_producto() + " Nombre: " +
s.getNombre() + " Distancia: " + s.getDistancia());

        } while (c.moveToNext());

        semillasEnSpinner = new String[semillas.size()];

        for (int j = 0; j < semillas.size(); j++) {
            semillasEnSpinner[j] = semillas.get(j).getNombre();
        }

    } else
        Log.e("@Sonia", "No se pudieron cargar las semillas ");

    return semillasEnSpinner;

}

public void guardarTerreno(String nombre, float areaDB, float largo,
float ancho) {

    Log.d("@Prueba", "Name = " + nombre + "Area: " + areaDB + " Largo:
" + largo + " Ancho: " + ancho);

    ContentValues terr = new ContentValues();

```

```
terr.put("nombre", nombre);
terr.put("area", areaDB);
terr.put("largo", largo);
terr.put("ancho", ancho);
DBcampo.insert("terreno", null, terr);

Cursor c = DB.hacer_consulta("SELECT * FROM terreno");

if (c.moveToFirst())
    Log.d("@Sonia", "Hay algo en la base terrenos :D, se inserto
correctamente");

else
    Log.d("@Sonia", "Aún no hay datos");

}

}
```

DISEÑO DE MÓDULO III: MÓDULO DE VISUALIZACIÓN DE MAPAS

CAPTURA	DESCRIPCIÓN
<p style="text-align: center;">DETALLES 196.38919 m</p>  <p>100.5281 6 m</p> <p>ÁREA DEL TERRENO: 200.0 m2 #PLANTAS A SEMBRAR: 2 DISTANCIA ENTRE PLANTAS 75.0cm</p> <p>¿CÓMO VA A SEMBRAR?</p> <p>LARGO ANCHO</p>	<p>Pantalla XI. En esta pantalla el usuario podrá visualizar los detalles del terreno, es decir, podrá conocer las dimensiones que tiene el espacio en la que desea sembrar, el área así como el total de plantas que puede sembrar y las distancias en las que deben sembrarse de acuerdo al producto escogido.</p>
 <p>196.38919 m</p> <p>100.5281 6 m</p> <p>ÁREA DEL TERRENO: 200.0 m2 #PLANTAS A SEMBRAR: 2 DISTANCIA ENTRE PLANTAS 75.0cm</p> <p>¿CÓMO VA A SEMBRAR?</p> <p>LARGO ANCHO</p> <p>CONTINUAR</p>	<p>La pantalla XI contendrá la opción de ¿cómo desea sembrar? en esta sección el usuario podrá escoger entre sembrar a lo largo o ancho de su terreno. será necesario escoger una forma de siembra y continuar.</p>

CÓDIGO DE MÓDULO III: MÓDULO DE VISUALIZACIÓN DE MAPAS

mostrar_calculos.java

```
package com.example.sonia.azadon;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

import static com.example.sonia.azadon.agregar.altura;
import static com.example.sonia.azadon.agregar.area;
import static com.example.sonia.azadon.agregar.base;
import static com.example.sonia.azadon.pasos.distanciaEntrePlantas;

public class mostrar_calculos extends AppCompatActivity implements
View.OnClickListener {

    /*Esta clase recolecta la información obtenida y la muestra en una
    pantalla
    antes de sembrar (plantas a sembrar, el área del terreno, lo largo,
    lo ancho,
    la distancia que ocupará cada planta*/

    Button btn_calculo_continuar;

    private TextView tv_areaTerreno;
    private TextView tv_num_plantas;
    private TextView tv_distancia_plantas;
    private TextView tv_calculos_base;
    private TextView tv_calculos_altura;
    private Button toggle_largo, toggle_ancho;
    public static String largoAncho = "largo";

    public static int numPlantas;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_mostrar_calculos);

        btn_calculo_continuar = (Button)
findViewById(R.id.btn_calculos_continuar);
        tv_areaTerreno = (TextView) findViewById(R.id.tv_areaterreno);
        tv_num_plantas = (TextView) findViewById(R.id.tv_num_plantas);
        tv_distancia_plantas = (TextView)
findViewById(R.id.tv_distancia_plantas);
        toggle_ancho = (Button) findViewById(R.id.toggle_ancho);
        toggle_largo = (Button) findViewById(R.id.toggle_largo);
        tv_calculos_altura = (TextView)
findViewById(R.id.tv_calculos_altura);
        tv_calculos_base = (TextView) findViewById(R.id.tv_calculos_base);
```

```

tv_calculos_base.setText(String.format("%.1f", base / 100) + " m");
tv_calculos_altura.setText(String.format("%.1f", altura / 100) + "
m");

toggle_largo.setOnClickListener(this);
toggle_ancho.setOnClickListener(this);

numPlantas = (int) (area / distanciaEntrePlantas);
//Calcula el número de plantas que podremos sembrar

tv_areaTerreno.setText(area + " m2");
tv_num_plantas.setText("" + numPlantas);
tv_distancia_plantas.setText(distanciaEntrePlantas + "cm");

btn_calculo_continuar.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View v) {
        Intent i = new Intent(mostrador_calculos.this,
sembrado.class);
        startActivity(i);
    }
});
}

@Override
public void onClick(View v) {
    switch (v.getId()) {
        /* Permite elegir entre sembrar a lo largo, o a lo ancho del
terreno*/
        case R.id.toggle_largo:
            largoAncho = "largo";

toggle_largo.setBackgroundColor(getResources().getColor(R.color.colorPrimar
yDark));

toggle_ancho.setBackgroundColor(getResources().getColor(R.color.colorAccent
));
            break;
        case R.id.toggle_ancho:
            largoAncho = "ancho";

toggle_ancho.setBackgroundColor(getResources().getColor(R.color.colorPrimar
yDark));

toggle_largo.setBackgroundColor(getResources().getColor(R.color.colorAccent
));
            break;
    }
}
}
}

```

DISEÑO DE MÓDULO IV: MÓDULO DE AVISO PARA SIEMBRA

CAPTURA	DESCRIPCIÓN
 <p>The screenshot shows a mobile application interface with a white background and a green header. At the top, it says "centro del terreno" in green. Below this is an illustration of three oranges. The text "Continúa hacia el frente 100 + 1/4 pasos" is displayed in green, followed by a black stick figure icon walking forward. Below that, it says "Repite por 2 gira a la derecha y camina 100 + 1/4 pasos" in green, accompanied by an illustration of a person kneeling and planting a sapling. At the bottom, it says "Repite desde el paso uno al tres por 2 veces mas, hasta haber completado el area del terreno" in green. A green button with the text "TERMINAR" is at the very bottom.</p>	<p>Pantalla XII.</p> <p>En esta pantalla se mostrará al usuario la distancia en la que deberá sembrar cada una de sus semillas y se le indicará la forma en la que debe medir sus pasos para poder sembrar en el lugar adecuado. Al oprimir el botón de Terminar la aplicación mostrará la pantalla I.</p>

CÓDIGO DE MÓDULO IV: MÓDULO DE AVISO PARA SIEMBRA

sembrado.java

```
package com.example.sonia.azadon;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

import static com.example.sonia.azadon.agregar.altura;
import static com.example.sonia.azadon.agregar.base;
import static com.example.sonia.azadon.mostrar_calculos.largoAncho;
import static com.example.sonia.azadon.pasos.distanciaEntrePlantas;
import static com.example.sonia.azadon.pasos.paso_promedio;

public class sembrado extends AppCompatActivity {

    private Button btn_sembrado_terminar;
    private TextView tv_sembrado_uno;
    private TextView tv_sembrado_dos;
    private TextView tv_sembrado_tres;
    private TextView tv_sembrado_cuatro;

    int pasos;
    int numplantasA, numplantasB;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        /*
        La clase por excelencia, una de las mas importantes de todas.
        Da al usuario instrucciones específicas sobre que debe de hacer
        que distancia debe dejar entre cada planta dada "en sus propios
        pasos"
        utilizando la información obtenida anteriormente

        */
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sembrado);

        btn_sembrado_terminar = (Button)
        findViewById(R.id.btn_sembrado_terminar);
        tv_sembrado_uno = (TextView) findViewById(R.id.tv_sembrado_uno);
        tv_sembrado_dos = (TextView) findViewById(R.id.tv_sembrado_dos);
        tv_sembrado_tres = (TextView) findViewById(R.id.tv_sembrado_tres);
        tv_sembrado_cuatro = (TextView)
        findViewById(R.id.tv_sembrado_cuatro);

        Log.d("@Sonia", "El paso promedio es: " + paso_promedio);
        Log.d("@Sonia ->", "Se recibe Base: " + base + " Altura: " +
        altura);
        Log.d("@Sonia", "La distancia entre plantas es : " +
```

```

distanciaEntrePlantas);

    /*Calcula cuántas veces tendremos que repetir el proceso
    Dependiendo de si elegimos sembrar a lo largo o a lo
    ancho*/

    if (largoAncho.equals("largo")) {
        numplantasA = (int) (base / distanciaEntrePlantas);
        numplantasB = (int) (altura / distanciaEntrePlantas);
        Log.d("@Sonia", "Plantas a lo largo " + numplantasA + " Ancho:
" + numplantasB);
    } else {
        numplantasA = (int) (altura / distanciaEntrePlantas);
        numplantasB = (int) (base / distanciaEntrePlantas);
        Log.d("@Sonia", "Plantas a lo largo " + numplantasB + " Ancho:
" + numplantasA);
    }

    /*Debido a que los mensajes son dinámicos, es decir
    no siempre son los mismos pasos ni la misma cantidad
    de fracciones, el texto debe ser tratado individualmente
    ya que es difícil decirle a una persona, camina 0.75% de
    tus pasos, por lo cual se emplea 3/4 de paso
    */
    tv_sembrado_uno.setText(mensaje_uno());
    tv_sembrado_dos.setText(mensaje_dos());
    tv_sembrado_tres.setText(mensaje_tres());
    tv_sembrado_cuatro.setText(mensaje_cuatro());

    btn_sembrado_terminar.setOnClickListener(new
View.OnClickListener() {
        @Override
        public void onClick(View v) {
            finish();
            Intent i = new Intent(sembrado.this, campo.class);
            startActivity(i);
        }
    });
}

/*Esta función convierte a lenguaje natural los decimales a fracciones
de pasos que la persona debe de dar
*/
public String calcularPasosRedondeo(float planta, float paso) {

    int a = (int) (planta / paso);
    double b = planta / paso;
    float c = (float) (b - a); //Obtener solo los decimales

    if (c <= 0.25)
        return "1/4";
    else if (c <= 0.5)
        return "1/2";
    else if (c <= 0.75)
        return "3/4";
    else
        return "1";
}

```

```

//Función que calcula si es un paso, son pasos o si debe aumentarse
public String calcularPasos () {

    String m = "";

    if ((int) (distanciaEntrePlantas / paso_promedio) < 0)
        m += " " + calcularPasosRedondeo (distanciaEntrePlantas,
paso_promedio) + " de paso";

    else if (calcularPasosRedondeo (distanciaEntrePlantas,
paso_promedio).equals("1"))
        m += " " + (int) (distanciaEntrePlantas / paso_promedio) + 1 +
" pasos";

    else
        m += " " + (int) (distanciaEntrePlantas / paso_promedio) + " +
" + calcularPasosRedondeo (distanciaEntrePlantas, paso_promedio) + "
pasos";

    return m;
}

public String mensaje_uno () {
    String m = "Camina hacia la derecha ";
    m += calcularPasos ();
    m += " y voltea hacia el centro del terreno";
    return m;
}

public String mensaje_dos () {
    String m = "Continúa hacia el frente ";
    m += calcularPasos ();
    return m;
}

public String mensaje_tres () {
    return " Repite por " + numplantasA + " gira a la derecha y camina
" + calcularPasos ();
}

public String mensaje_cuatro () {
    return "Repite desde el paso uno al tres por " + (numplantasB) + "
veces mas, hasta haber completado el area del terreno";
}
}

```