

Plataforma de Gestión de Datos de Sensores para Agricultura de Precisión

Martina Nahman

SMARTAGRO

Es un sistema que recopila, almacena, analiza y visualiza datos provenientes de sensores instalados en cultivos y suelos agrícolas. Su objetivo es ayudar a los agricultores a tomar decisiones informadas sobre riego, fertilización, control de plagas y otras prácticas agrícolas, optimizando la producción y reduciendo costos.





01 Integración

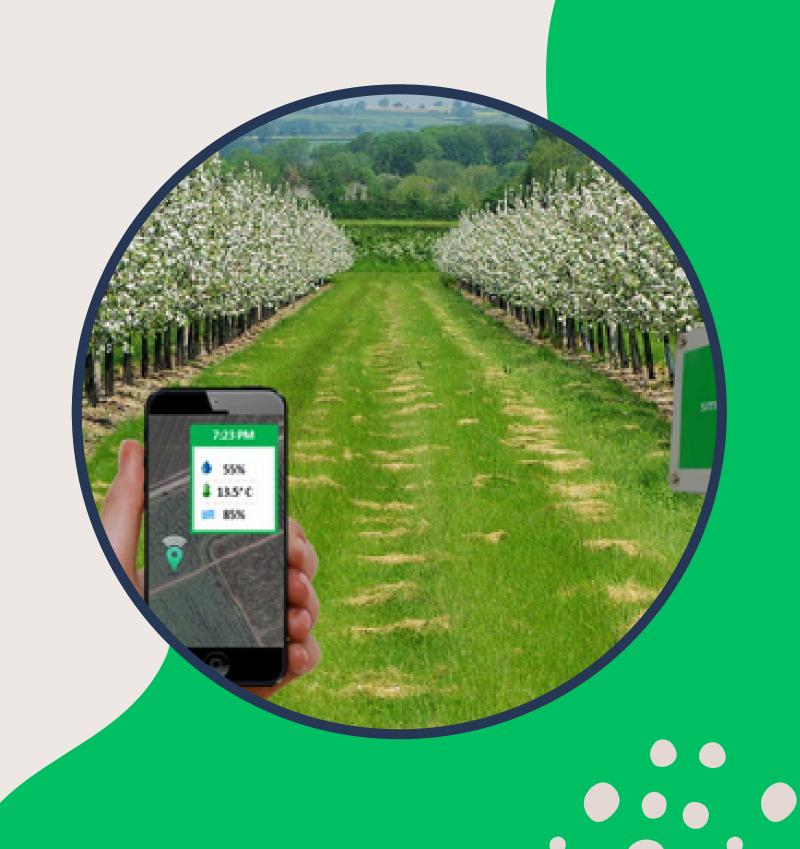
Integración de datos en tiempo real en un sólo lugar 02
Registro

Registro de datos con menor margen de error 03
Interfaz

Interfaz intuitiva para la gestión de la información

REQUISITOS FUNCIONALES

- Gestión de sensores y datos agrícolas
- Módulo ABM
- Autenticación de usuarios



REQUISITOS

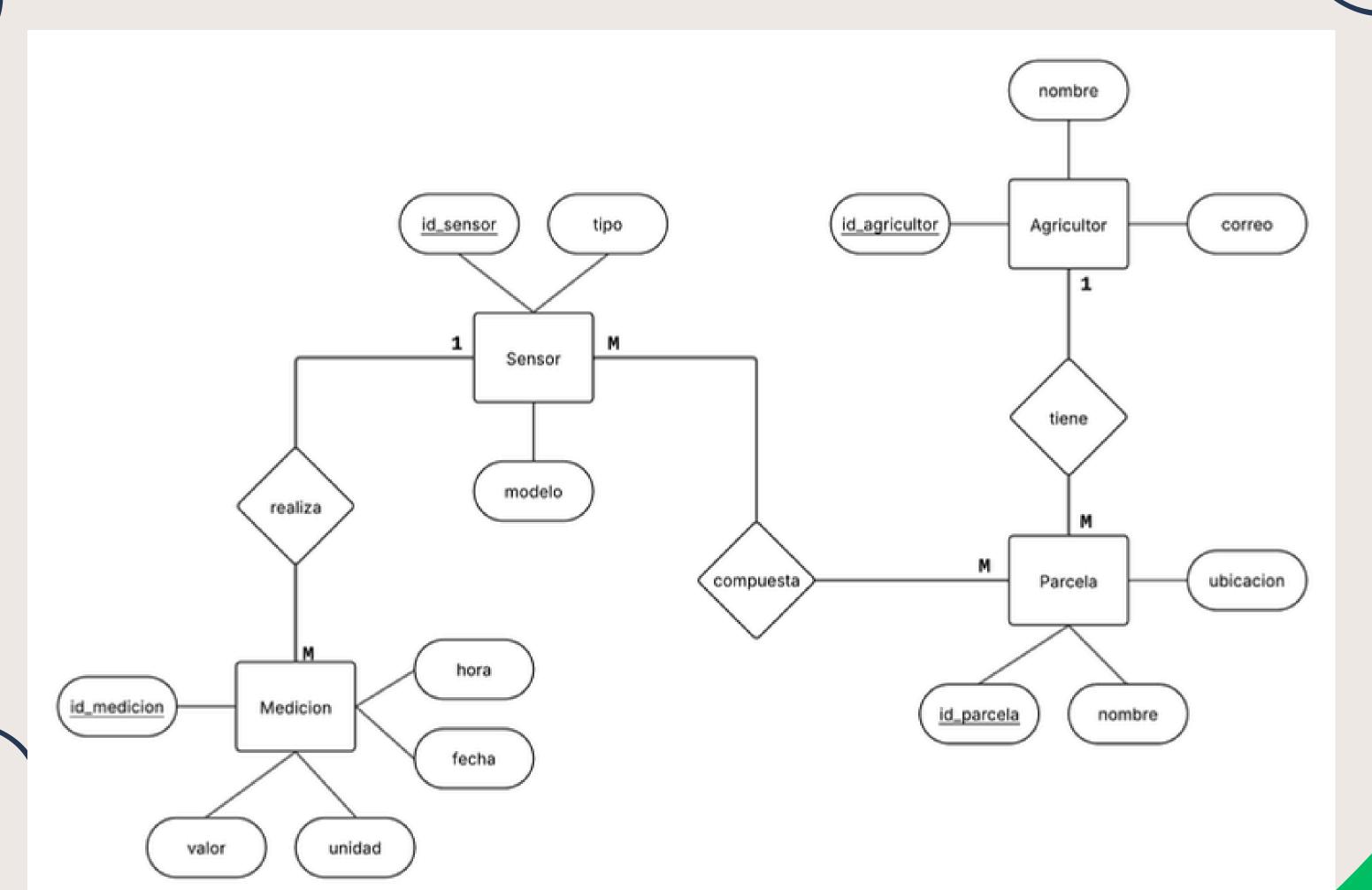
NO FUNCIONALES

Seguridad y control de acceso

2 Usabilidad



DIAGRAMA ENTIDAD RELACIÓN



MODELO RELACIONAL

Agricultor(id_agricultor, nombre, correo)

Parcela(id_parcela, nombreParcela, ubicacion, id_agricultor)

Sensor(id_sensor, tipo, modelo)

Medicion(id_medicion, hora, fecha, valor, unidad,id_sensor)

Compuesta(id_parcela, id_sensor)

IMPLEMENTACIÓN DEL SISTEMA



PostgreSQL



Java



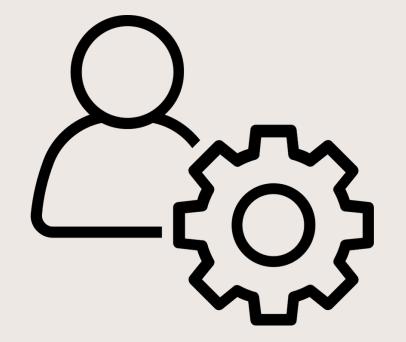
Java Swing



Arquitectura MVC

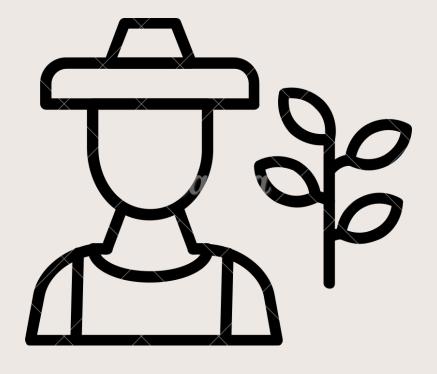
ROLES





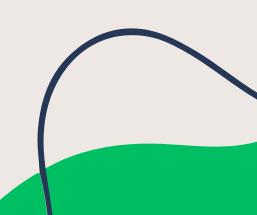
Acceso completo:

Acceso a todas las tablas y
 ABM de las mismas



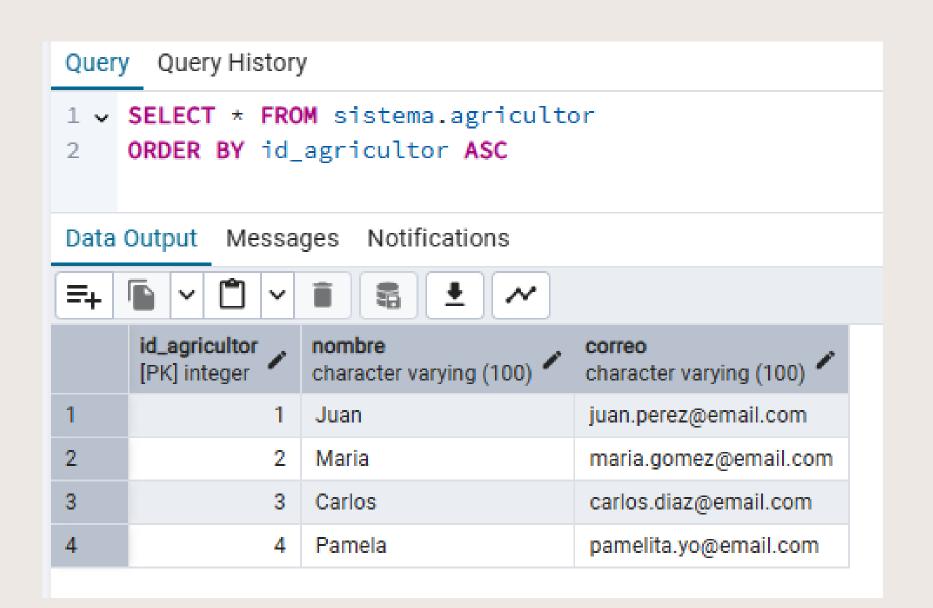
Acceso limitado:

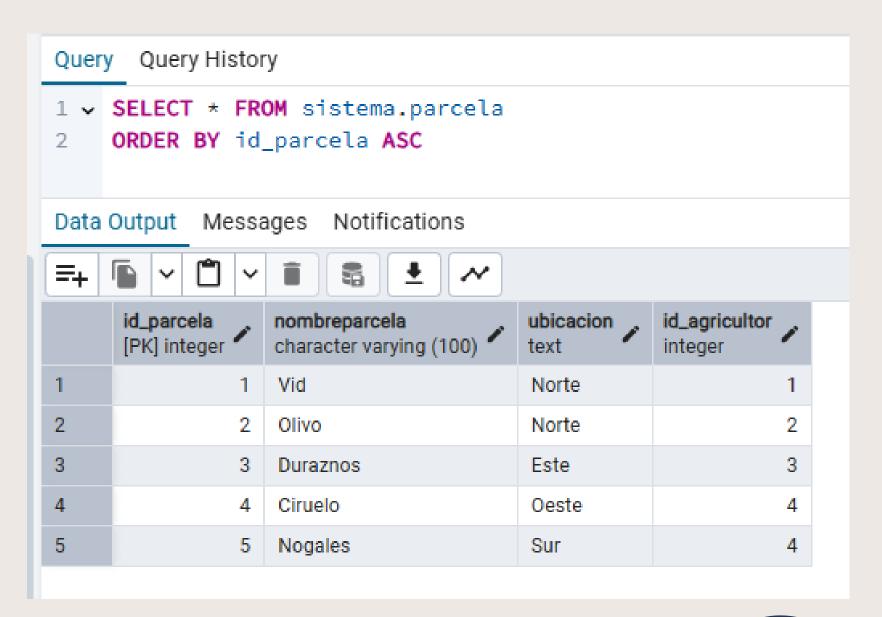
- Acceso de lectura a sensores y parcelas
- Acceso total (ABM) sobre medición













Query Query History

- 1 SELECT * FROM sistema.medicion
- ORDER BY id_medicion ASC

Data Output Messages Notifications

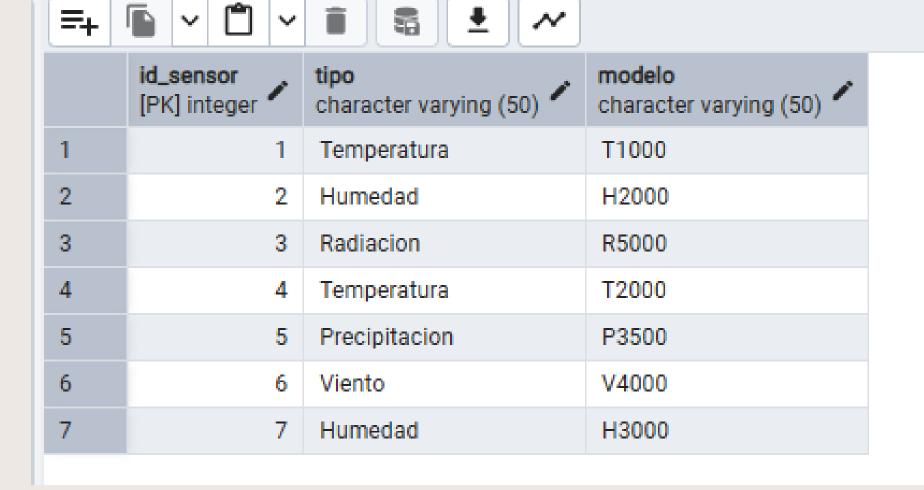
= +			Showing rows: 1 to 7					
	id_medicion [PK] integer	hora time without time zone	fecha date	valor numeric (10,2)	unidad character varying (20)	id_sensor integer		
1	1	08:00:00	2025-06-10	22.50	°C	1		
2	2	08:05:00	2025-06-10	78.30	%	2		
3	3	08:10:00	2025-06-10	540.00	W/m²	3		
4	4	08:15:00	2025-06-10	25.10	°C	4		
5	5	08:20:00	2025-06-10	12.70	mm	5		
6	6	08:25:00	2025-06-10	18.40	km/h	6		
7	7	08:30:00	2025-06-10	65.90	%	7		



Query Query History

- 1 v SELECT * FROM sistema.sensor
- ORDER BY id_sensor ASC

Data Output Messages Notifications



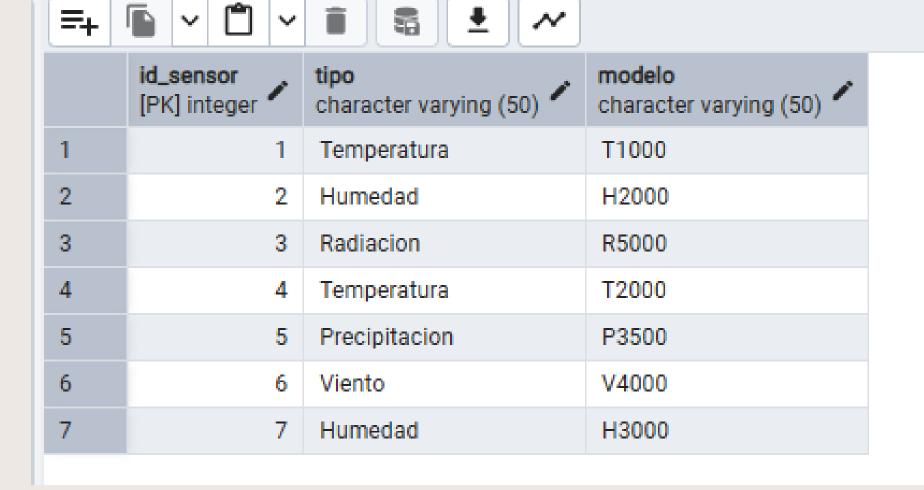
Query Query History								
1 ∨ SELECT * FROM sistema.compuesta								
2 ORDER BY id_parcela ASC, id_sensor ASC								
Data Output Messages Notifications								
=+								
	id_parcela [PK] integer	id_sensor [PK] integer						
1	1	1						
2	1	2						
3	2	3						
4	3	4						
5	3	5						
6	4	6						
7	5	7						



Query Query History

- 1 v SELECT * FROM sistema.sensor
- ORDER BY id_sensor ASC

Data Output Messages Notifications



Query Query History								
1 ∨ SELECT * FROM sistema.compuesta								
2 ORDER BY id_parcela ASC, id_sensor ASC								
Data Output Messages Notifications								
=+								
	id_parcela [PK] integer	id_sensor [PK] integer						
1	1	1						
2	1	2						
3	2	3						
4	3	4						
5	3	5						
6	4	6						
7	5	7						

ELECCIÓN ABM



```
@Entity
@Table(name = "Agricultor", schema = "sistema")
public class Agricultor {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "id agricultor")
    private int id;
    @Column(nullable = false, length = 100)
    private String nombre;
    @Column(nullable = false, unique = true, length = 100)
    private String correo;
    @Column(nullable = false)
    private boolean activo;
    @OneToMany(mappedBy = "agricultor", cascade = CascadeType.ALL, orphanRemoval = true)
    private List<Parcela> parcelas;
```

ELECCIÓN ABM



```
@Entity
@Table(name = "Parcela", schema = "sistema")
public class Parcela {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "id parcela")
    private int id:
    @Column(name = "nombreParcela", nullable = false, length = 100)
    private String nombreParcela;
    @Column(nullable = false, columnDefinition = "TEXT")
    private String ubicacion;
    @Column(nullable = false)
   private boolean activo;
    @ManyToOne
    @JoinColumn(name = "id agricultor", nullable = false)
    private Agricultor agricultor;
```





🙆 SmartAgro - Menú Principal	-	×
& Gestionar Agricultores		
Cestionar Parcelas		
Exportar PDF		
£ Exportar Base de Datos		
≛ Importar Base de Datos		
X Cerrar sesión		

CONCLUSIÓN

SmartAgro es una solución práctica y eficiente para la gestión de datos agrícolas. Desarrollado con Java, Swing y PostgreSQL, ofrece una interfaz clara y funcionalidades clave que facilitan el trabajo en el campo y permiten futuras mejoras tecnológicas.



MUCHAS GRACIAS

