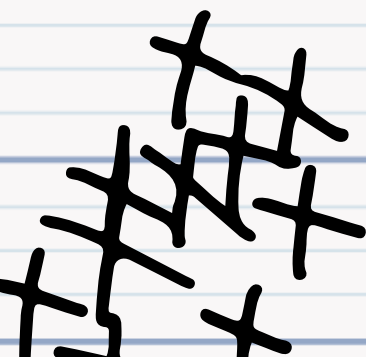
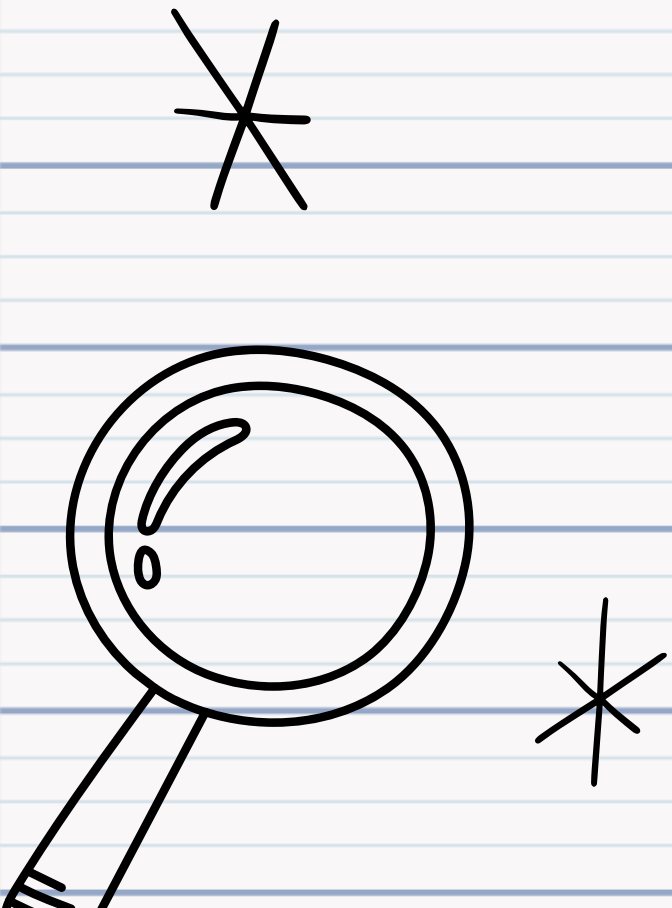
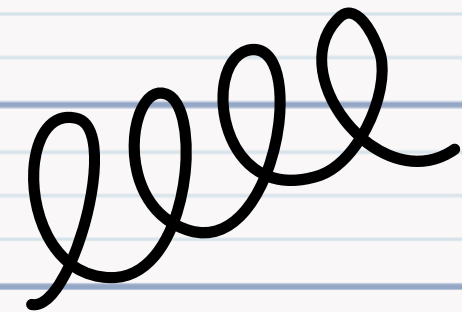
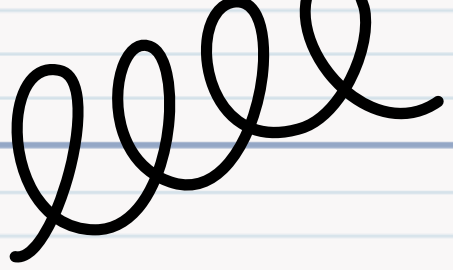


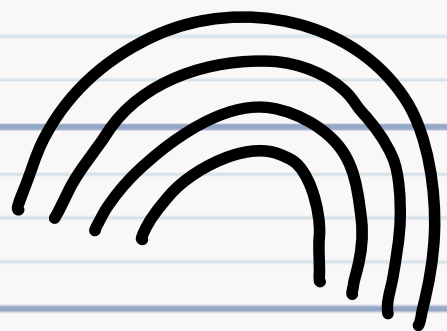
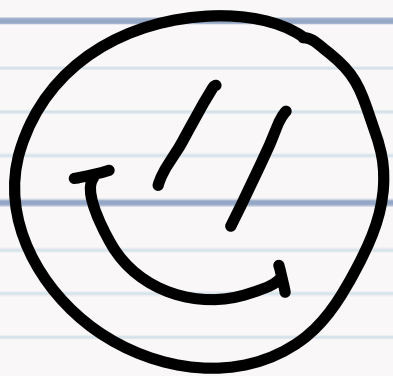
BASE DE DATOS

BY: T.Irvin





¿QUE ES UNA BASE DE DATOS?



Una base de datos es una recopilación organizada de información o datos estructurados, que normalmente se almacena de forma electrónica en un sistema informático. Normalmente, una base de datos está controlada por un sistema de gestión de bases de datos (DBMS). En conjunto, los datos y el DBMS, junto con las aplicaciones asociadas a ellos, reciben el nombre de sistema de bases de datos, abreviado normalmente a simplemente base de datos.

TIPOS DE BASES DE DATOS:

```
graph TD; A[TIPOS DE BASES DE DATOS:] --> B[RELACIONALES]; A --> C[NO RELACIONALES]; B --> D["• Utilizan un modelo de datos relacional.  
• La información se organiza en tablas con filas y columnas.  
• Ejemplos de DBMS relacionales incluyen MySQL, PostgreSQL, y Oracle."]; C --> E["• No siguen el modelo relacional.  
• Pueden usar modelos como documentos, grafos, clave-valor, etc.  
• MongoDB, Cassandra y Redis son ejemplos de bases de datos NoSQL."];
```

RELACIONALES

- Utilizan un modelo de datos relacional.
- La información se organiza en tablas con filas y columnas.
- Ejemplos de DBMS relacionales incluyen MySQL, PostgreSQL, y Oracle.

NO RELACIONALES

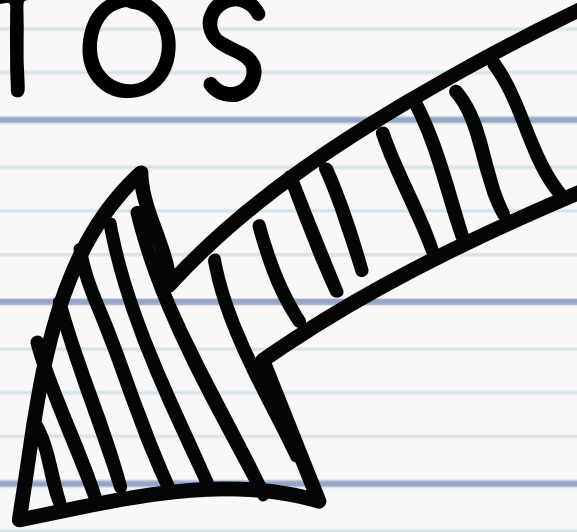
- No siguen el modelo relacional.
- Pueden usar modelos como documentos, grafos, clave-valor, etc.
- MongoDB, Cassandra y Redis son ejemplos de bases de datos NoSQL.

OTROS TIPOS DE BASES DE DATOS

Bases de Datos Orientadas a Objetos

Bases de Datos Jerárquicas

Bases de Datos de Red

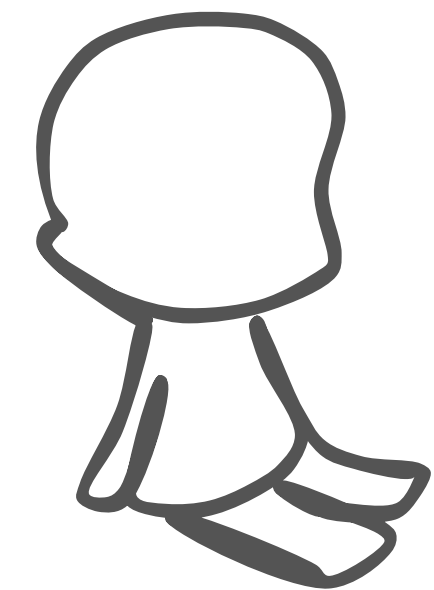


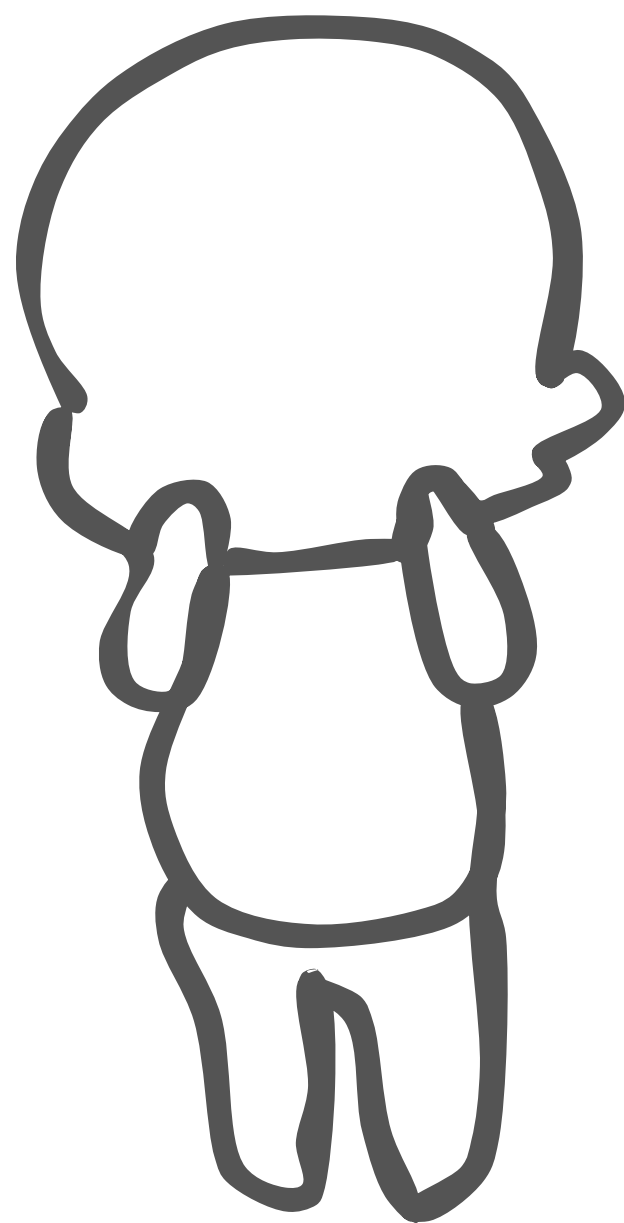
DBMS Relacionales:

- MySQL: Ampliamente utilizado, de código abierto.
- Oracle Database: Potente y escalable.
- Microsoft SQL Server: Desarrollado por Microsoft, compatible con plataformas Windows.

DBMS NoSQL:

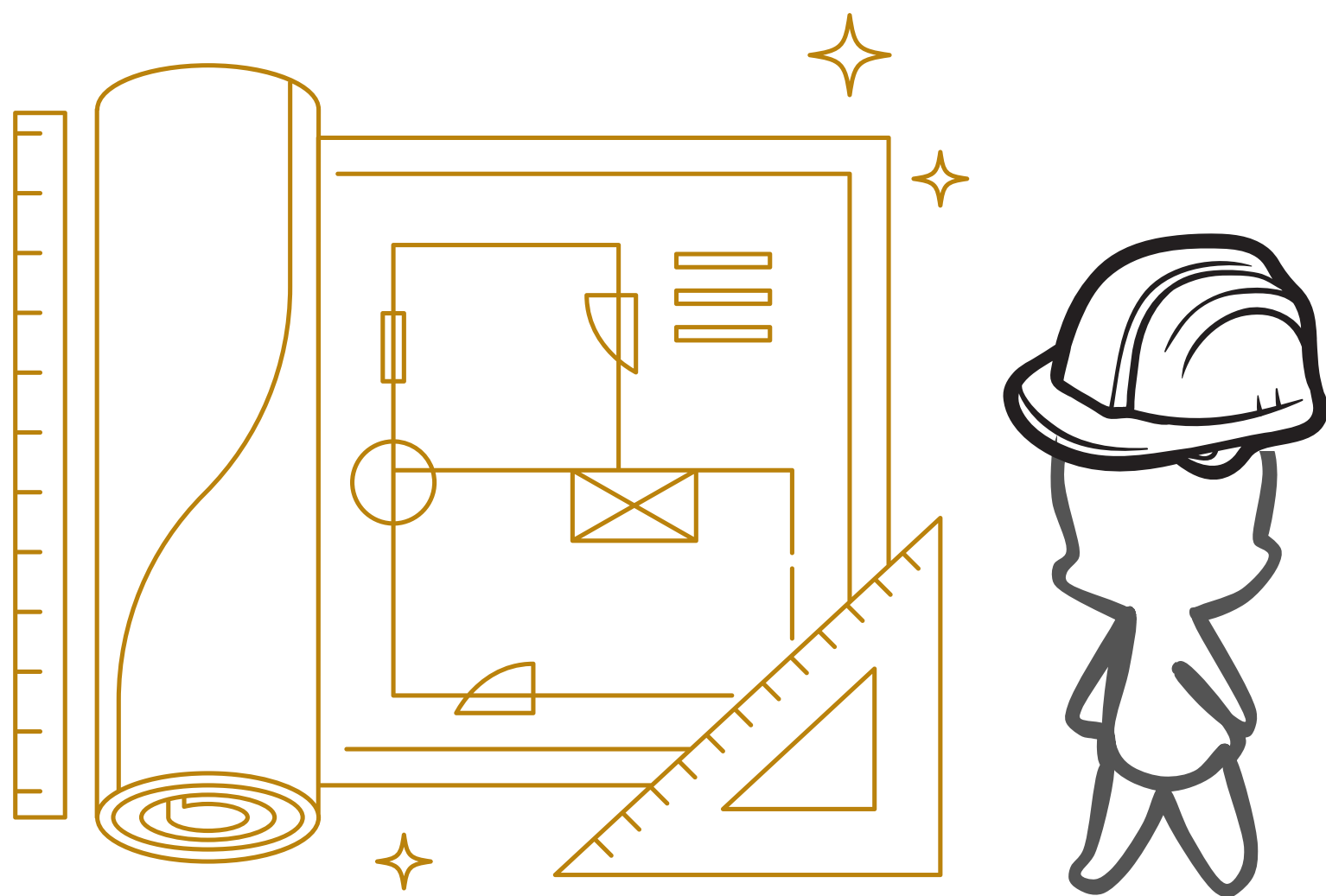
- MongoDB: Almacena datos en documentos BSON (formato binario JSON).
- Cassandra: Escalable y diseñado para manejar grandes cantidades de datos distribuidos.
- Redis: Almacenamiento en memoria de estructuras de datos clave-valor.



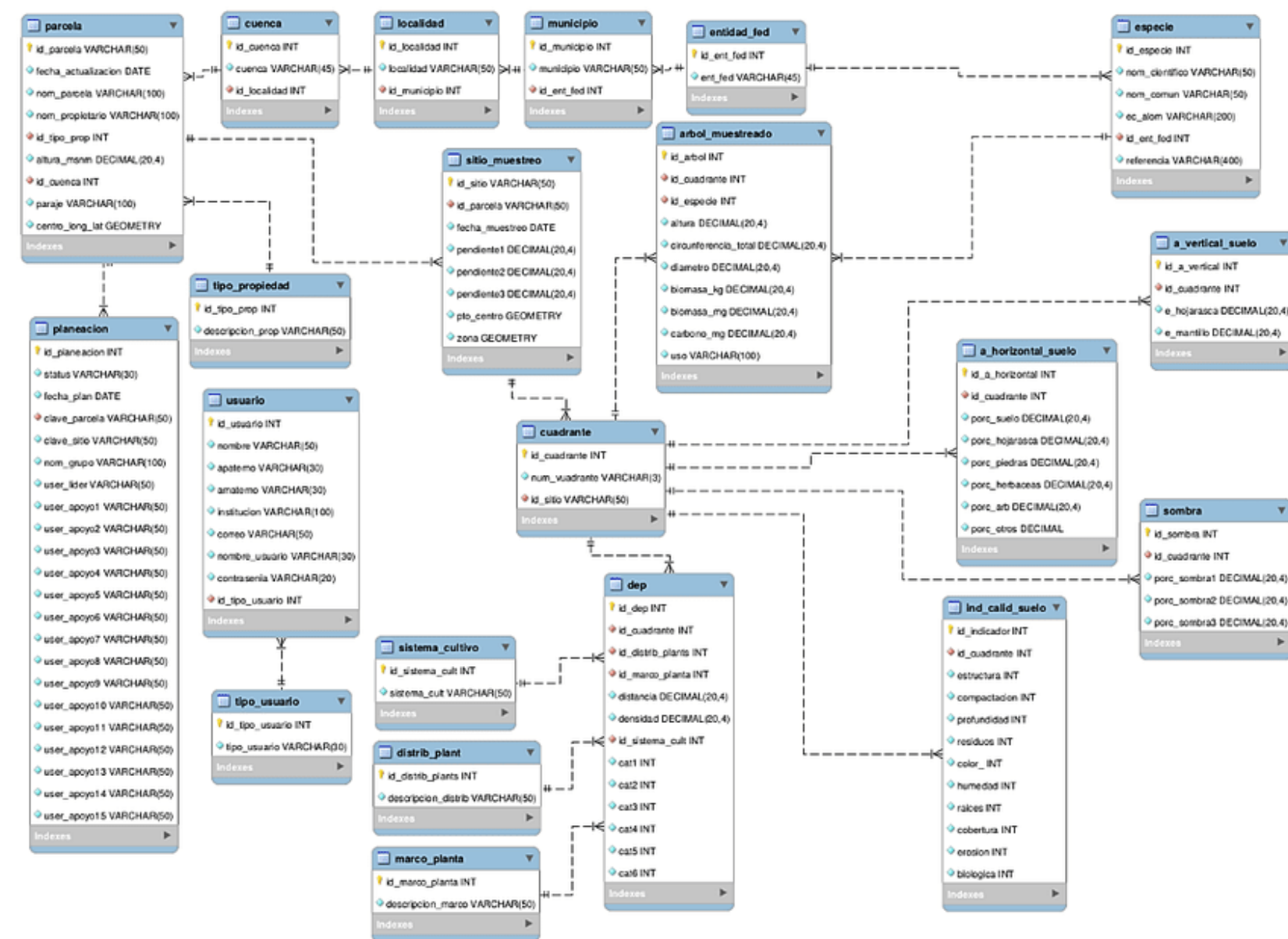




Pero antes hay que
aprender teoria



Asi como un
arquitecto diseña
planos

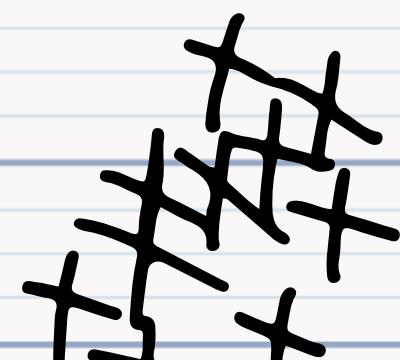
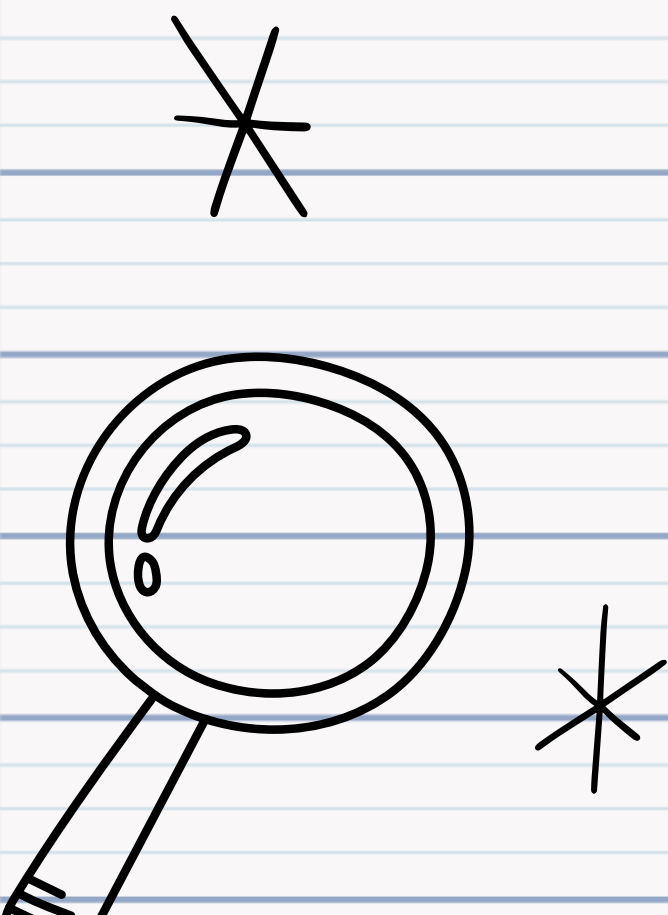
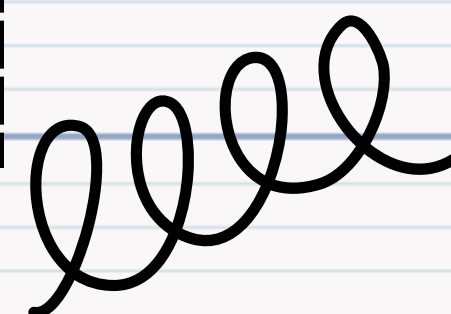


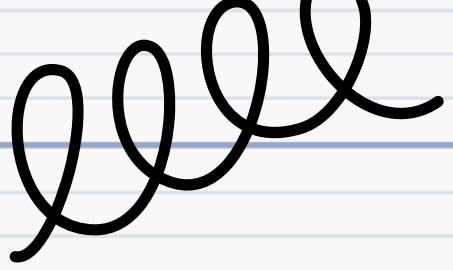
un arquitecto de bases de
datos
diseña base de datos



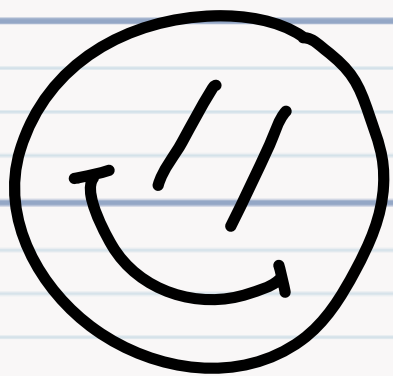
DIAGRAMAS DE BASES DE DATOS

Entidad relación





BASES DE DATOS RELACIONALES



Las bases de datos relacionales son un tipo de base de datos que sigue el modelo de datos relacional. Este modelo organiza los datos en tablas bidimensionales, donde la información se almacena en filas y columnas. Cada fila en una tabla representa una entidad única y cada columna representa un atributo específico de esa entidad. Estas tablas están relacionadas entre sí mediante claves primarias y foráneas.

HABLEMOS EL MISMO IDIOMA

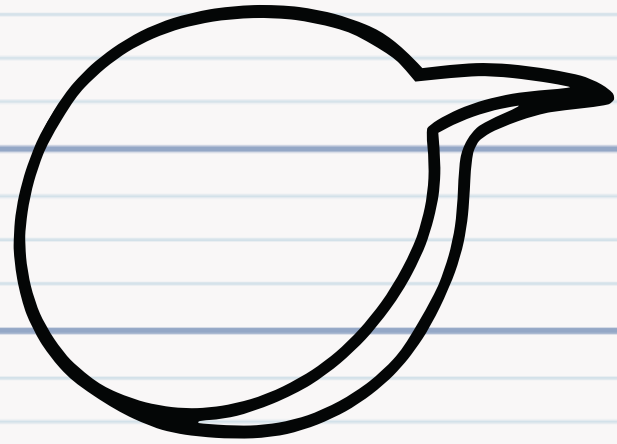
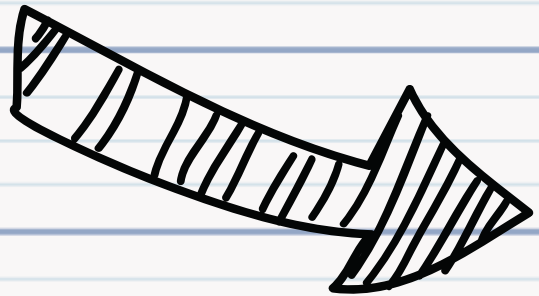


Tabla: Una tabla es la unidad básica de almacenamiento en una base de datos relacional. Cada tabla tiene un nombre único y está compuesta por filas y columnas.

Fila: Cada fila en una tabla representa una instancia única de la entidad que la tabla describe. Cada fila tiene un valor único en la columna de la clave primaria, que sirve para identificar de manera única esa fila.

Columna: Cada columna en una tabla representa un atributo específico de la entidad. Las columnas tienen un nombre único y un tipo de dato asociado.

Clave Primaria: Es un campo (o conjunto de campos) en una tabla que tiene valores únicos en cada fila. Se utiliza para identificar de manera única cada registro en la tabla.



Clave Foránea: Es un campo en una tabla que establece una relación con la clave primaria de otra tabla. Sirve para vincular las filas de una tabla con las de otra.

Relación: La relación entre dos tablas se establece mediante la comparación de las claves primarias y foráneas. Puede ser de uno a uno, uno a muchos o muchos a muchos.

Integridad Referencial: Es la propiedad que garantiza que las relaciones entre las tablas son válidas y que no se pueden crear referencias a registros inexistentes.

Lenguaje SQL (Structured Query Language): Se utiliza para interactuar con bases de datos relacionales. SQL proporciona comandos para realizar operaciones como la inserción, actualización, consulta y eliminación de datos.



Entidad:

Una entidad es un objeto o concepto del mundo real que se puede distinguir y reconocer.

En el diseño de bases de datos, una entidad se representa como una tabla.

Cada fila en la tabla representa una instancia o registro específico de esa entidad.

Ejemplos de entidades pueden incluir "Estudiante", "Producto", "Empleado", etc.

Atributo:

Un atributo es una propiedad o característica específica que describe una entidad.

Cada columna en la tabla de una entidad representa un atributo.

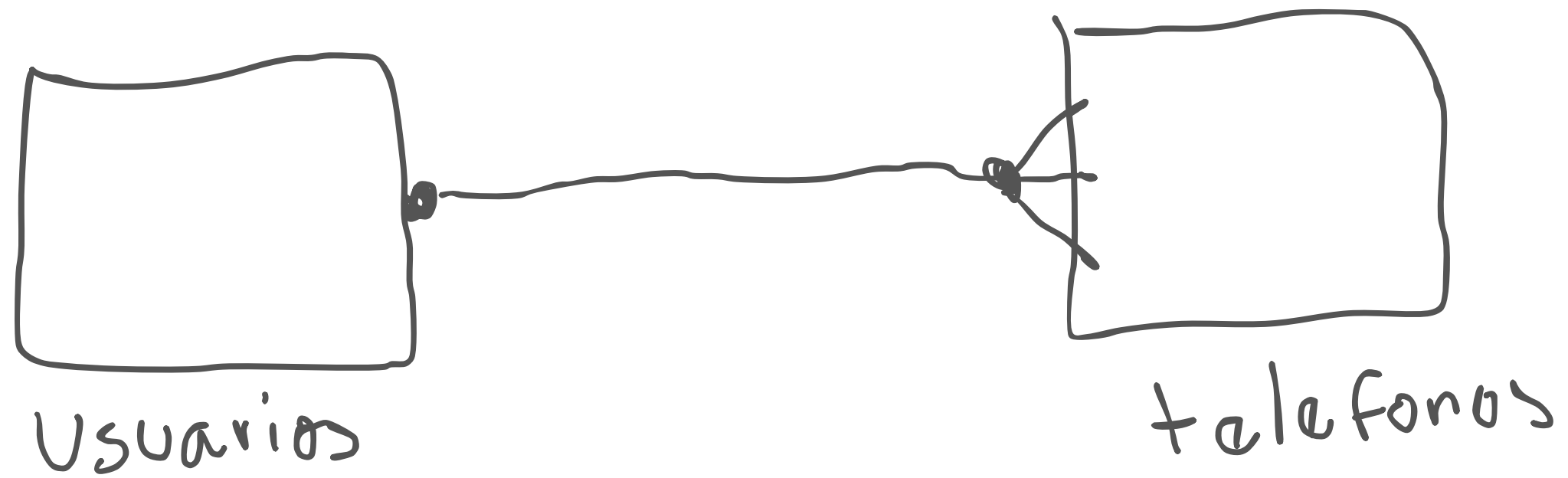
Los atributos contienen información sobre las instancias individuales de la entidad.

Ejemplos de atributos para la entidad "Estudiante" podrían ser "Nombre", "Edad", "Dirección", etc.

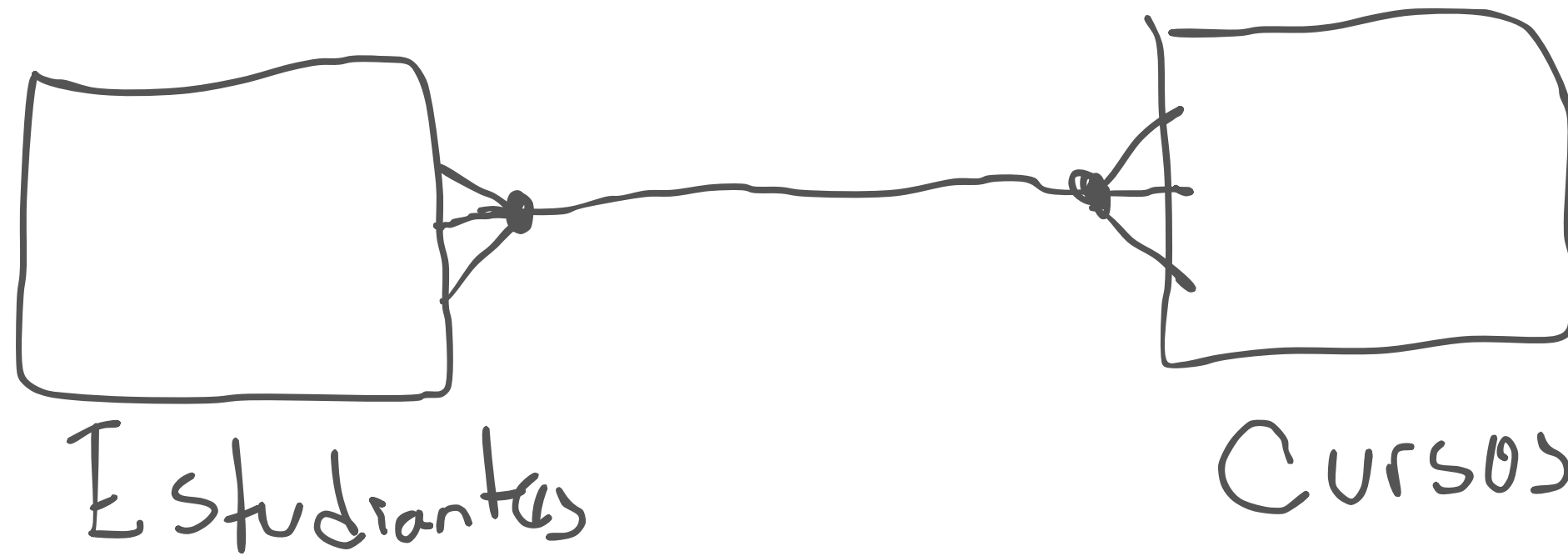
RELACIONES UNO A UNO



RELACIONES UNO A MUCHOS



RELACIONES MUCHOS A MUCHOS



Clientes

ID_Cliente

Nombre

Email

Telefono

Ordenes

ID_Orden

Fecha

Total

ID_Cliente

TABLA CLIENTES			
ID_Cliente	Nombre	Email	Teléfono
1	John Doe	john@example.com	555-1234
2	Jane Smith	jane@example.com	555-5678
TABLA ORDENES			
ID_Orden	Fecha	Total	ID_Cliente
101	1/20/2022	150	1
102	1/21/2022	200	2

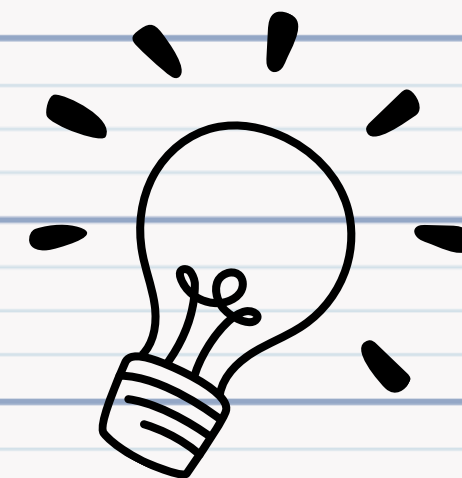
Diseña una base de datos para un sistema escolar que incluya información sobre estudiantes, profesores, asignaturas y calificaciones. Considera las siguientes entidades y sus atributos:

Estudiantes (ID_Estudiante, Nombre, Edad, Dirección, etc.)

Profesores (ID_Profesor, Nombre, Especialidad, etc.)

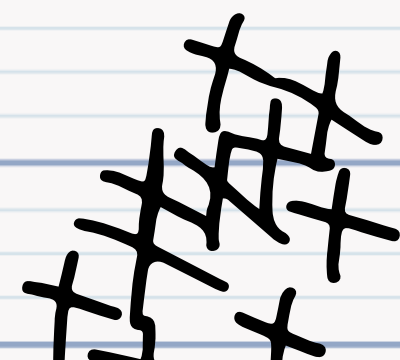
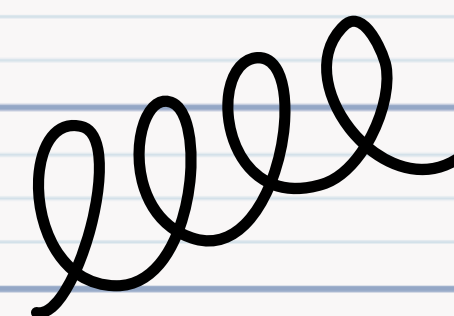
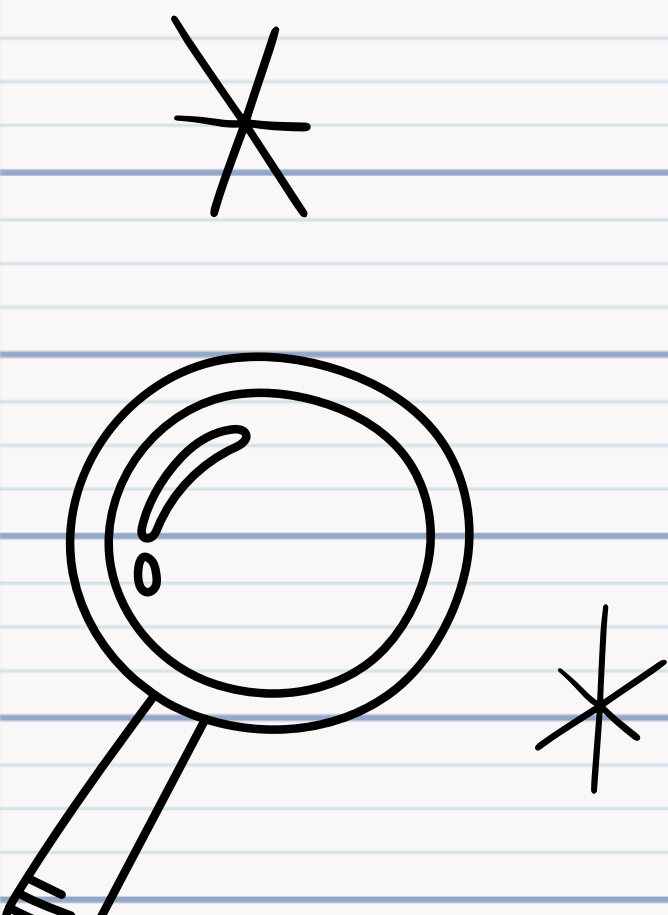
Asignaturas (ID_Asignatura, Nombre, Créditos, etc.)

Calificaciones (ID_Calificacion, ID_Estudiante, ID_Asignatura, Nota, Fecha, etc.)



PARTE II

Tipos de datos



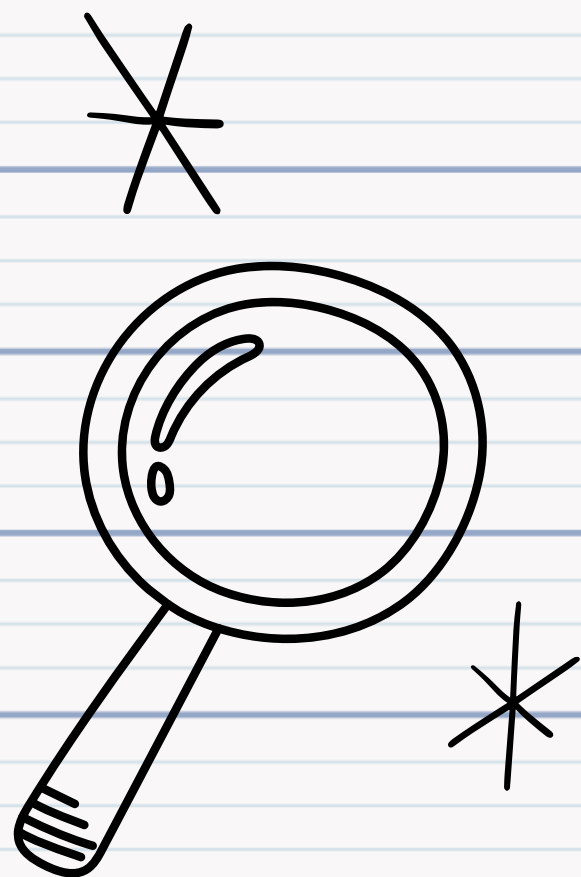
Tipo de Datos	Descripción
INT	Números enteros, por ejemplo, 1, 10, -5
BIGINT	Números enteros más grandes que INT
FLOAT	Números de punto flotante, por ejemplo, 3.14, -0.001
DOUBLE	Números de punto flotante de mayor precisión que FLOAT
DECIMAL(p, s)	Números decimales con una precisión específica (p) y escala (s)
VARCHAR(n)	Cadenas de texto de longitud variable con una longitud máxima de n
CHAR(n)	Cadenas de texto de longitud fija con una longitud exacta de n
TEXT	Cadenas de texto de longitud variable (longitud máxima depende del sistema)
DATE	Fechas en formato 'YYYY-MM-DD'
TIME	Tiempos en formato 'HH:MM:SS'
DATETIME	Fechas y tiempos en formato 'YYYY-MM-DD HH:MM:SS'
BOOLEAN	Almacenamiento de valores de verdad (TRUE/FALSE)

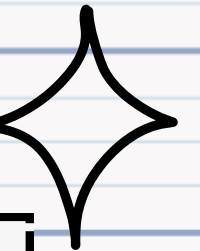
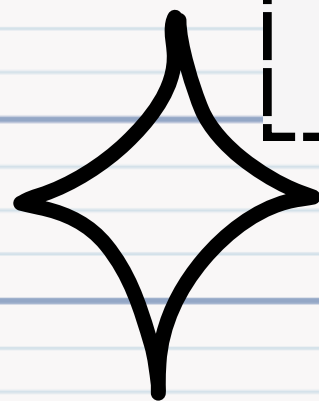
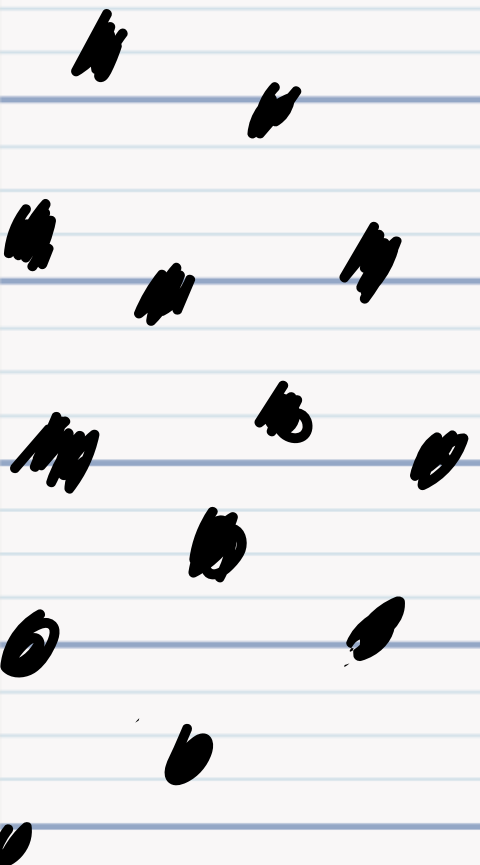
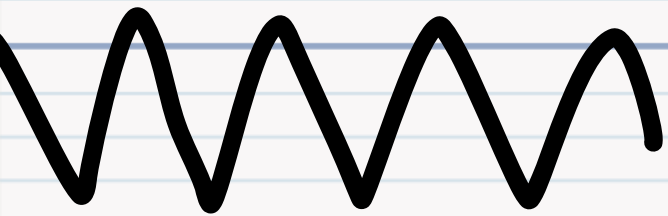


```
1 CREATE TABLE usuarios (  
2     id INT,  
3     nombre VARCHAR(50),  
4     edad INT,  
5     saldo DECIMAL(10, 2),  
6     activo BOOLEAN  
7 );
```

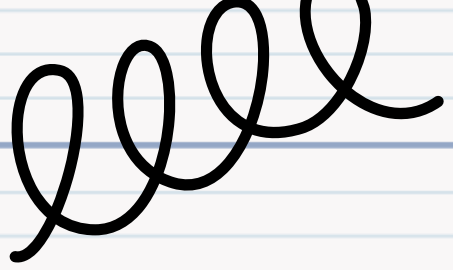


BONUS

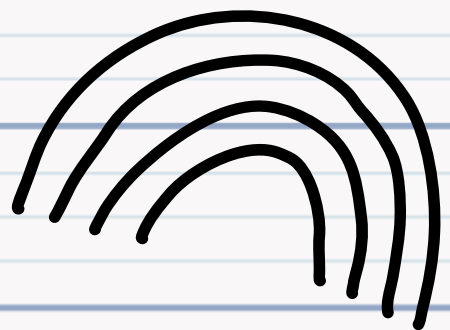
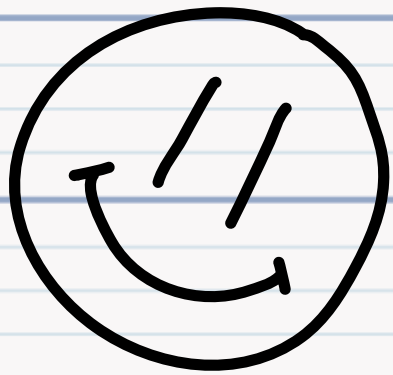
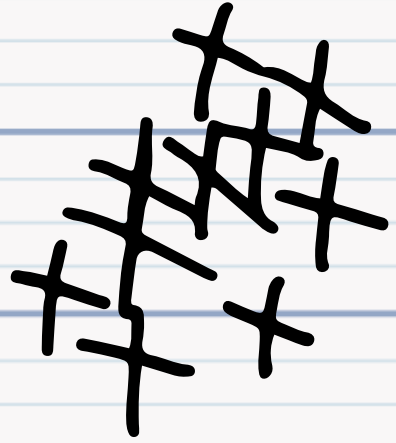




AGILE

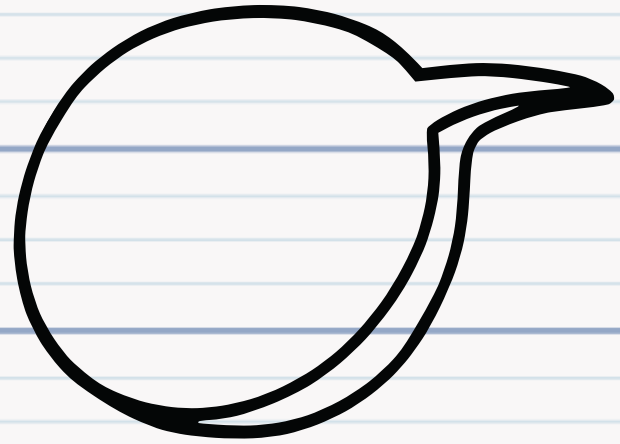


SCRUM



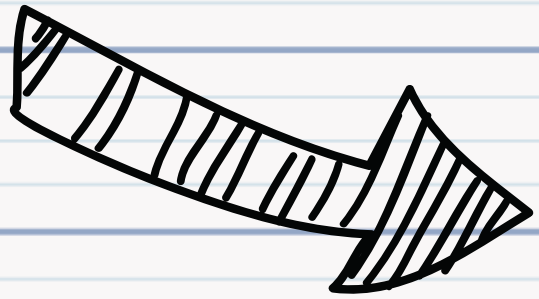
Scrum es un marco de trabajo ágil utilizado en el desarrollo de proyectos, especialmente en software. Se centra en la colaboración, la flexibilidad y la entrega incremental de productos. En Scrum, se dividen los proyectos en iteraciones llamadas "sprints", que suelen durar de 2 a 4 semanas, durante las cuales se desarrolla y entrega un conjunto de funcionalidades prioritarias. El proceso implica equipos autoorganizados, revisiones regulares y ajustes continuos para adaptarse a los cambios.





GLOSARIO

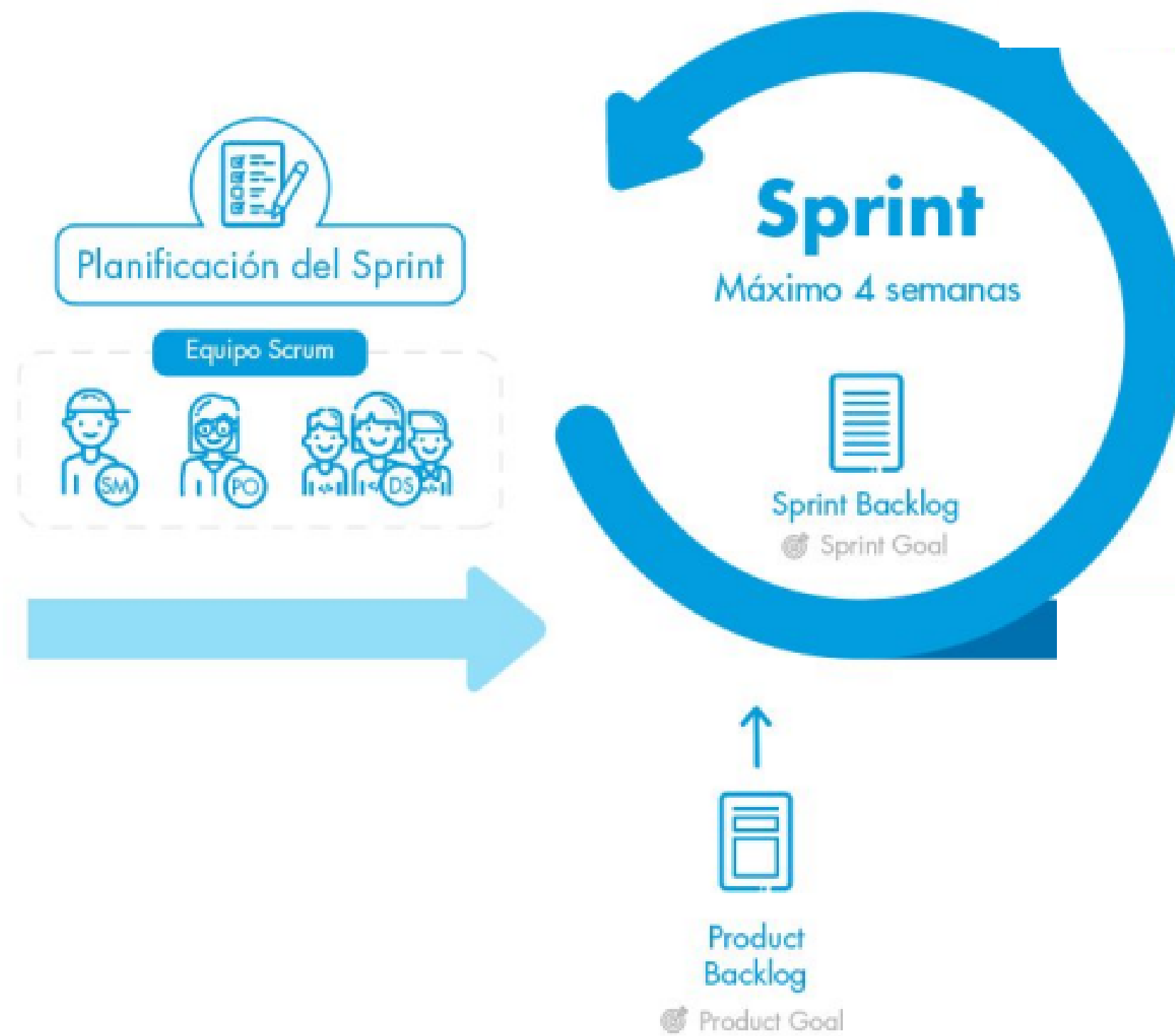
Product Owner (Dueño del Producto): Define las características del producto y prioriza el trabajo del equipo.

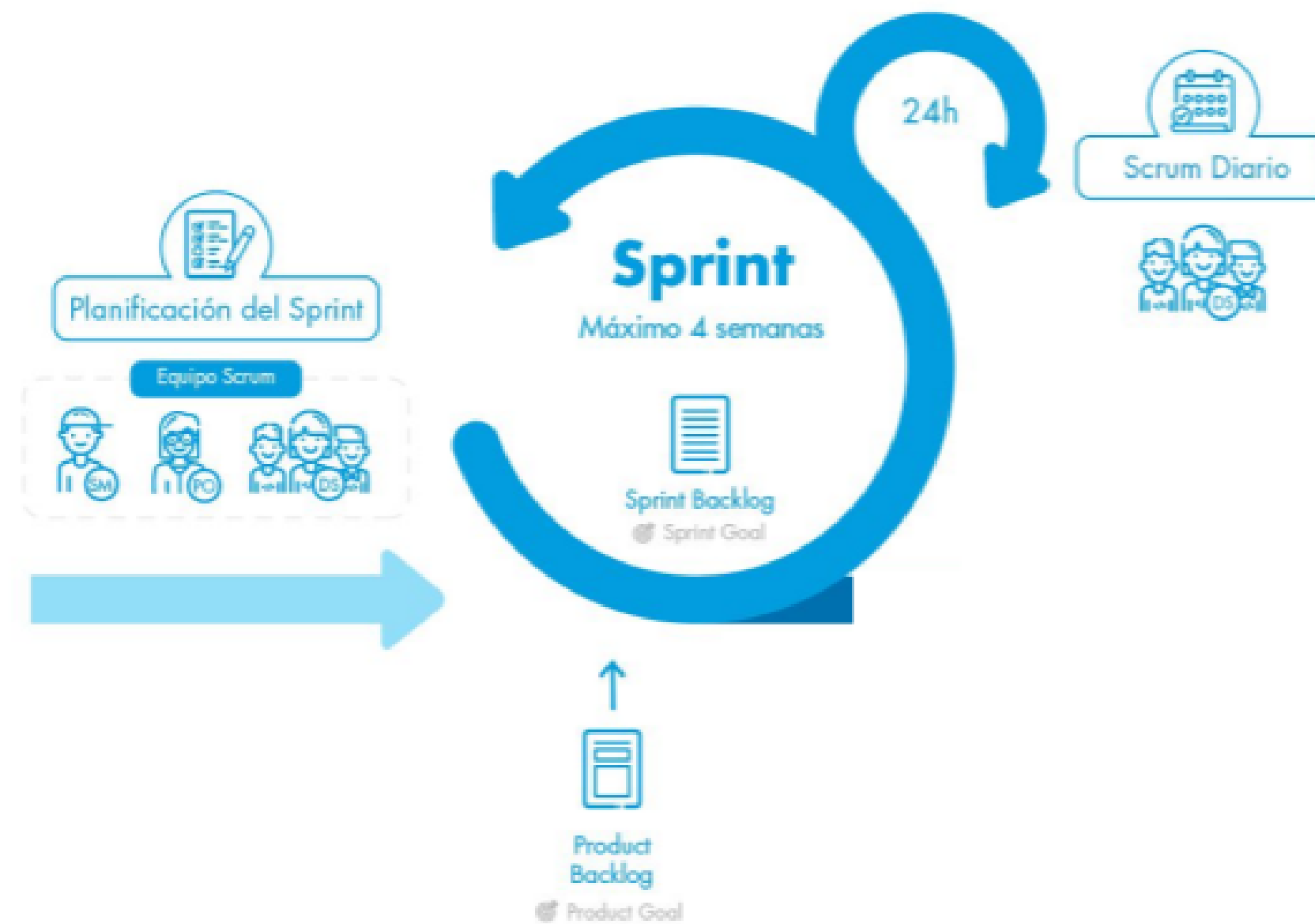


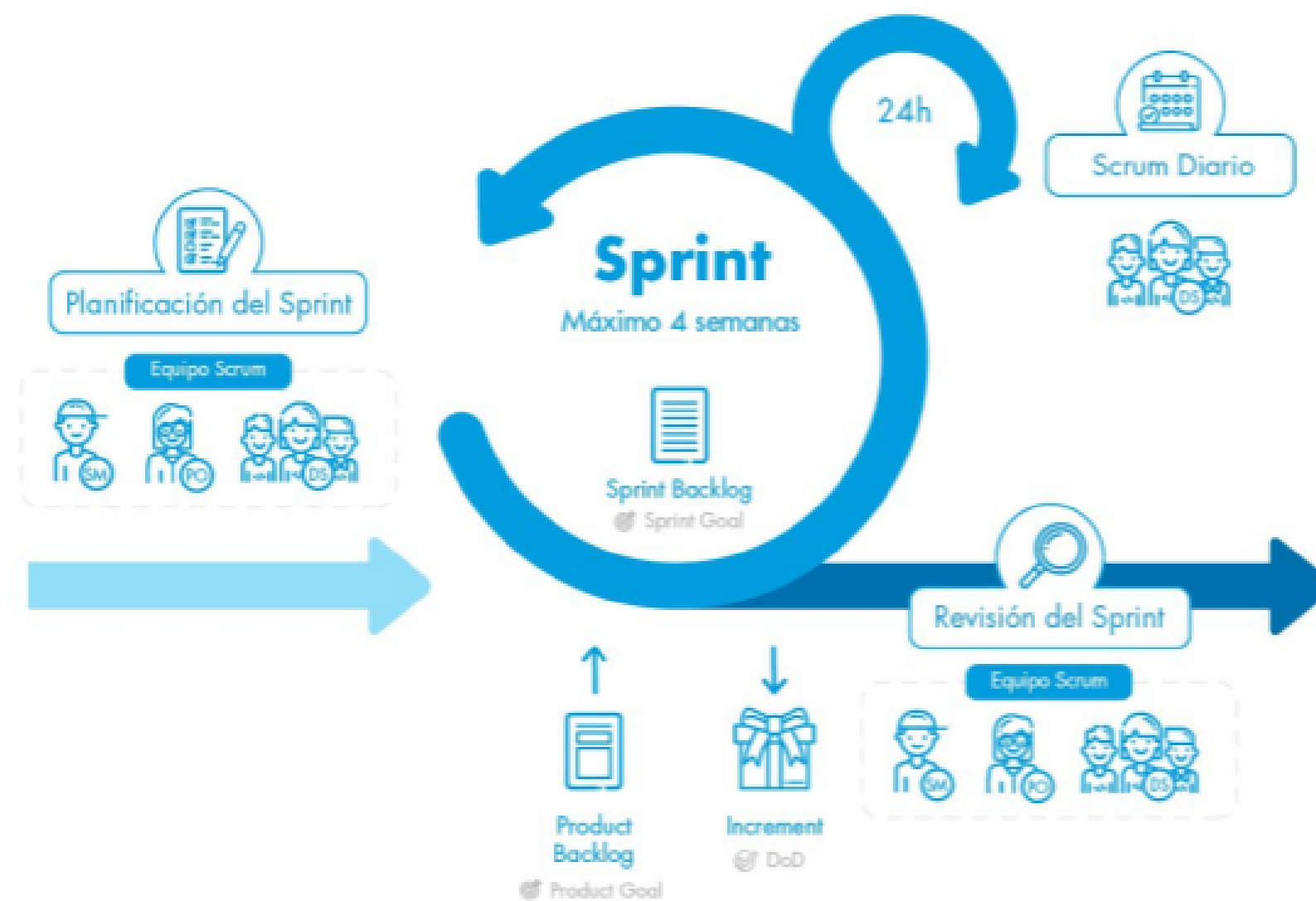
Scrum Master: Facilita el proceso Scrum, elimina obstáculos y trabaja para mejorar la eficiencia del equipo.

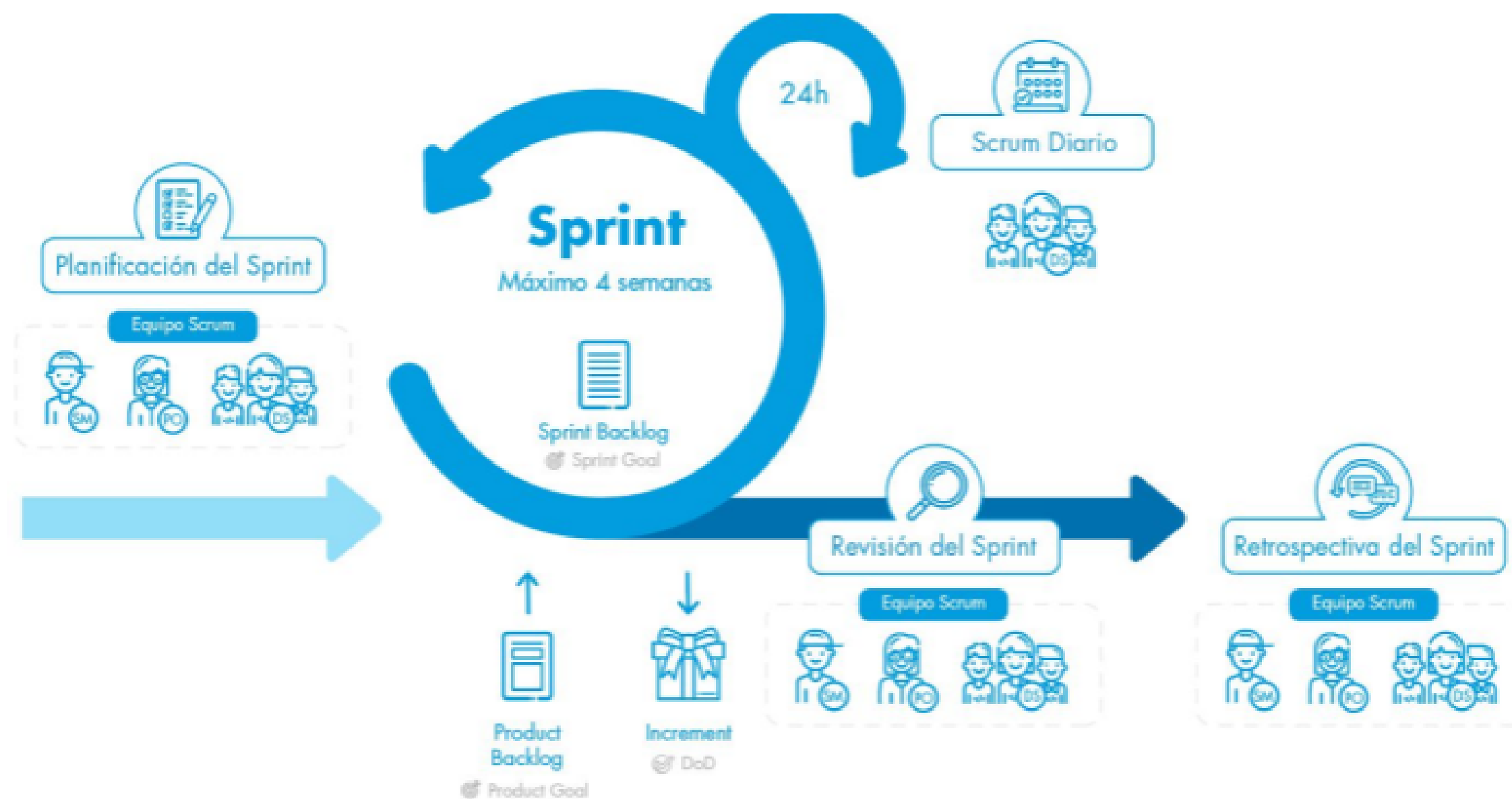


Equipo de Desarrollo: Profesionales que construyen el producto, son autónomos y colaboran para cumplir con los objetivos del Sprint.









1. Definir el Producto:

- Comienza por identificar y definir claramente el producto que estás desarrollando. Esto puede ser un software, un servicio o cualquier otro producto.

2. Crear el Backlog del Producto:

- El Product Owner (dueño del producto) es responsable de crear y gestionar el Product Backlog. Este es una lista priorizada de todas las características, mejoras y tareas necesarias para el producto.

3. Planificación del Sprint:

- Un Sprint es un periodo de tiempo fijo durante el cual se desarrolla un conjunto de características del Product Backlog. Antes de cada Sprint, el equipo de desarrollo y el Product Owner se reúnen en una reunión de planificación del Sprint para seleccionar las tareas que se abordarán en el próximo Sprint.

4. Reunión Diaria (Daily Scrum):

- Durante el Sprint, el equipo tiene reuniones diarias cortas llamadas Daily Scrum. En estas reuniones, cada miembro del equipo informa sobre lo que hizo ayer, lo que planea hacer hoy y si hay algún obstáculo en su camino. El objetivo es mantener a todos en la misma página y abordar cualquier problema de inmediato.

5. Desarrollo del Producto:

- Durante el Sprint, el equipo trabaja en las tareas seleccionadas del Product Backlog. El objetivo es entregar un incremento de producto funcional al final del Sprint.

6. Revisión del Sprint:

- Al final de cada Sprint, hay una reunión de revisión del Sprint. El equipo presenta lo que se ha logrado durante el Sprint y recibe retroalimentación del Product Owner y otros interesados.

7. Retrospectiva del Sprint:

- Después de la revisión del Sprint, el equipo tiene una reunión de retrospectiva del Sprint. En esta reunión, se discuten los aspectos que funcionaron bien, los problemas encontrados y cómo mejorar en el próximo Sprint.

8. Actualizar el Product Backlog:

- Basándose en la retroalimentación y la revisión del Sprint, el Product Owner actualiza el Product Backlog. Se pueden agregar, eliminar o re-priorizar elementos según las necesidades del proyecto.

9. Repetir el Ciclo:

- El equipo repite este ciclo en cada Sprint. Cada Sprint tiene una duración fija, generalmente entre 1 y 4 semanas. Este ciclo de desarrollo iterativo y incremental continúa hasta que se alcanza el objetivo del proyecto o se decide interrumpir el desarrollo.