



DOCUMENTO DE BASE DE DATOS

PROYECTO FINAL

4o semestre Año 2022



INFORMACIÓN DEL DOCUMENTO

Nombre de Proyecto: Proyecto Final		
Preparada por:	Diaz, Ariadna Lopez, Federico Torena, Nahuel Vazquez, Christofer Santana, Joaquín	Fecha: 24/ 09 / 2022
Revisado por:		Fecha revisión:

Versiones

Ver. No.	Fecha Ver.	Actualizado por:	Descripción
1.0	24/09/2022	The Boys	Sprint 1



CONTENIDO

INTRODUCCIÓN	4
1. PROPÓSITO	4
2. ALCANCE	4
BASE DE DATOS	5
1. Introducción	5
1.1 ¿Qué es?	5
1.2 ¿Cómo funciona?	5
2. Estructura	5
2.1 Lista de tablas y sus estructuras	5
3. LISTA DE TABLAS Y SUS ESTRUCTURAS	7
4. LISTA DE RESTRICCIONES	10
5. LISTA DE SECUENCIAS DEFINIDAS	12
6. DIAGRAMA DE TABLAS Y RELACIONES	13
BASE DE DATOS CORPORATIVA	14
1. Introducción	14
1.1 ¿Qué es?	14
1.2 ¿Por qué es útil la BDC?	14
2. Tablas	14
2.1 Definición	14
2.1 Tabla de hechos	14
2.2 Tabla de dimensión	14
3. Tablas y relaciones de la base de datos corporativa.	15
PROCESO ETL	24
1. Introducción	24
1.1 ¿Qué es ETL?	24
2. Proceso ETL	24
Power BI	25
1. Introducción	25
1.1 ¿Qué es Power BI?	25
1.2 Beneficios	25
2. Power BI	25
2.1 Pestañas	25
2.2 Filtros	26
2.3 Visual	26
2.4 Imágenes de las pestañas	27

INTRODUCCIÓN

Este documento representará toda la documentación pertinente al trabajo realizado por los integrantes del equipo The Boys para el área de base de datos.

1. Propósito

Documentar el trabajo realizado en la base de datos del Proyecto de Desarrollo y Testing y agregar la nueva documentación pertinente al Proyecto final con el fin de cumplir con los requerimientos solicitados en cada fase del proyecto.

2. Alcance

Se espera contar con una base de datos y una base de datos corporativa final que funcione con los requerimientos solicitados por el cliente. Cumpliendo con los requerimientos establecidos. La meta principal por cumplir es brindar un producto final que sea utilizado para almacenar la información de campo requerida y poder extraerla para su posterior análisis.

BASE DE DATOS

1. Introducción

1.1 ¿Qué es?

Una base de datos es un programa que almacena una gran cantidad de datos de manera organizada para ser usados posteriormente.

1.2 ¿Cómo funciona?

Esta almacena los datos deseados de forma organizada, que después, por medio de una aplicación, puede utilizarse dichos datos.

En nuestro caso, la base de datos almacena los datos relacionados con las terneras y guacheras, para posteriormente utilizar y almacenar dichos datos por medio de aplicaciones web y móvil

2. Estructura

2.1 Lista de tablas y sus estructuras

Está conformada por un total de 12 tablas, para poder manejar los datos de la estructura.

Las tablas que se van a utilizar son:

- Tabla de enfermedad, donde se tiene un registro de las enfermedades
- Tabla de guachera, donde se guarda la información de las mismas
- Tabla de alimentos, donde están los datos de los tipos de alimentos para las terneras
- Tabla de medicamento, donde se tiene los datos de los medicamentos
- Tabla de terneras, donde están los datos sobre las terneras
- Tabla vive, donde están los datos de entrada y salida de una ternera al nacer
- Tabla padece, donde están los datos de la enfermedad que padece la ternera
- Tabla suministra, donde están los datos de qué medicamento se le suministra a una ternera.
- Tabla de alimento de la ternera, este contiene los datos de los alimentos que se le proporcionaron a la ternera
- Tabla de registro del peso de ternera, donde se almacena los datos sobre los diferentes pesos que se registran lo largo de la vida de la ternera
- Tabla de registro de la temperatura de ternera, donde se tiene las diferentes temperaturas que puede tener la ternera
- Tabla de usuarios, que almacena los distintos usuarios que tiene el sistema

3. Lista de tablas y sus estructuras

❖ LATABLA TABLE_NAME ❖
1 La Tabla ENFERMEDAD;
2 La Tabla GUACHERA;
3 La Tabla ALIMENTO;
4 La Tabla MEDICAMENTO;
5 La Tabla TERNERA;
6 La Tabla VIVE;
7 La Tabla PADECE;
8 La Tabla SUMINISTRA;
9 La Tabla ALIM_TERNERA;
10 La Tabla REGISTRO_TERNERA_PESO;
11 La Tabla REGISTRO_TERNERA_TEMPERATURA;
12 La Tabla USUARIOS;

❖ COLUMN_NAME	❖ DATA_TYPE	❖ NULLABLE
1 ID_USUARIO	NUMBER(38,0)	No
2 NOMBRE	VARCHAR2(20 BYTE)	No
3 APELLIDO	VARCHAR2(20 BYTE)	No
4 NOMBRE_USUARIO	VARCHAR2(20 BYTE)	No
5 CONTRASENIA	VARCHAR2(50 BYTE)	No
6 TIPO	VARCHAR2(20 BYTE)	Yes
7 ESTADO	NUMBER(38,0)	Yes
8 CEDULA	VARCHAR2(20 BYTE)	Yes
9 TITULO	VARCHAR2(20 BYTE)	Yes
10 CANT_HORAS	NUMBER(38,0)	Yes

❖ COLUMN_NAME	❖ DATA_TYPE	❖ NULLABLE
1 ID_VIVE	NUMBER(38,0)	No
2 FECHA_ENTRADA	DATE	Yes
3 FECHA_SALIDA	DATE	Yes
4 ID_TERNERA	NUMBER(38,0)	Yes
5 ID_GUACHERA	NUMBER(38,0)	Yes

❖	COLUMN_NAME	❖	DATA_TYPE	❖	NULLABLE
1	ID_TERNERA		NUMBER(38,0)		No
2	CARAVANA_SNIG		VARCHAR2(20 BYTE)		Yes
3	ID_GUACHERA		NUMBER(38,0)		No
4	ID_PADRE		NUMBER(38,0)		Yes
5	ID_MADRE		NUMBER(38,0)		Yes
6	FECHA_NTO		TIMESTAMP(6)		Yes
7	PESO_AL_NACER		NUMBER(38,0)		Yes
8	RAZA		VARCHAR2(20 BYTE)		Yes
9	TIPO_DE_PARTO		VARCHAR2(20 BYTE)		Yes
10	ESTADO		NUMBER(38,0)		Yes

❖	COLUMN_NAME	❖	DATA_TYPE	❖	NULLABLE
1	ID_SUMINISTRA		NUMBER(38,0)		No
2	FECHA		DATE		Yes
3	COSTO_SUMINISTRO		NUMBER(38,0)		Yes
4	ID_MEDICAMENTO		NUMBER(38,0)		Yes
5	ID_TERNERA		NUMBER(38,0)		Yes
6	DOSIS		VARCHAR2(20 BYTE)		Yes

❖	COLUMN_NAME	❖	DATA_TYPE	❖	NULLABLE
1	ID_REGISTRO_TT		NUMBER(38,0)		No
2	TEMPERATURA		NUMBER(38,0)		No
3	FECHA		DATE		No
4	CARAVANA_SNIG		VARCHAR2(20 BYTE)		No
5	NOMBRE_USUARIO		VARCHAR2(20 BYTE)		No

❖	COLUMN_NAME	❖	DATA_TYPE	❖	NULLABLE
1	ID_REGISTRO_TP		NUMBER(38,0)		No
2	PESO		NUMBER(38,0)		No
3	FECHA		DATE		No
4	CARAVANA_SNIG		VARCHAR2(20 BYTE)		No
5	NOMBRE_USUARIO		VARCHAR2(20 BYTE)		No

❖	COLUMN_NAME	❖	DATA_TYPE	❖	NULLABLE
1	ID_PADECE		NUMBER(38,0)		No
2	FECHA_COMIENZO		DATE		Yes
3	FECHA_FINALIZADA		DATE		Yes
4	ID_ENFERMEDAD		NUMBER(38,0)		Yes
5	ID_TERNERA		NUMBER(38,0)		Yes

❖	COLUMN_NAME	DATA_TYPE	NULLABLE
1	ID_MEDICAMENTO	NUMBER (38,0)	No
2	NRO_MEDICAMENTO	NUMBER (38,0)	No
3	PRODUCTO	VARCHAR2 (25 BYTE)	Yes
4	MARCA	VARCHAR2 (20 BYTE)	Yes
5	COSTO	NUMBER (38,0)	Yes
6	DOSIS	NUMBER (38,0)	Yes
7	TIPO	VARCHAR2 (32 BYTE)	Yes
8	ESTADO	NUMBER (38,0)	Yes

❖	COLUMN_NAME	DATA_TYPE	NULLABLE
1	ID_GUACHERA	NUMBER (38,0)	No
2	UBICACION	VARCHAR2 (20 BYTE)	Yes
3	LARGO	NUMBER (38,0)	Yes
4	ANCHO	NUMBER (38,0)	Yes
5	CAPACIDAD	NUMBER (38,0)	Yes
6	ESTADO	NUMBER (38,0)	Yes

❖	COLUMN_NAME	DATA_TYPE	NULLABLE
1	ID_ENFERMEDAD	NUMBER (38,0)	No
2	VARIANTE	NUMBER (38,0)	Yes
3	NOMBRE	VARCHAR2 (20 BYTE)	Yes
4	TRATAMIENTO	VARCHAR2 (50 BYTE)	Yes
5	ESTADO	NUMBER (38,0)	Yes

❖	COLUMN_NAME	DATA_TYPE	NULLABLE
1	ID_ALIMENTO	NUMBER (38,0)	No
2	NOMBRE	VARCHAR2 (20 BYTE)	Yes
3	COSTO	NUMBER (38,0)	Yes
4	CANTIDAD	NUMBER (38,0)	Yes
5	ESTADO	NUMBER (38,0)	Yes
6	MARCA	VARCHAR2 (20 BYTE)	Yes

❖	COLUMN_NAME	DATA_TYPE	NULLABLE
1	ID ALIM_TERNERA	NUMBER (38,0)	No
2	ID_TERNERA	NUMBER (38,0)	Yes
3	ID_ALIMENTO	NUMBER (38,0)	Yes
4	COSTO ALIM	NUMBER (38,0)	Yes
5	FECHA	DATE	Yes



4. Lista de restricciones

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_OWNER	R_TABLE_NAME	R_CONSTRAINT_NAME
1 FK_ID_ALIMENTO ALIM	Foreign_Key	(null)	PDT	ALIMENTO	PK_ID_ALIMENTO
2 FK_ID_TERNERA ALIM	Foreign_Key	(null)	PDT	TERNERA	PK_ID_TERNERA
3 PK_ID ALIM TERNERA	Primary_Key	(null)	(null)	(null)	(null)
4 SYS_C005106	Check	"ID ALIM TERNERA" IS NOT NULL	(null)	(null)	(null)

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_OWNER	R_TABLE_NAME	R_CONSTRAINT_NAME
1 PK_ID ALIM	Primary_Key	(null)	(null)	(null)	(null)
2 SYS_C005083	Check	"ID ALIM" IS NOT NULL	(null)	(null)	(null)

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_OWNER	R_TABLE_NAME	R_CONSTRAINT_NAME
1 PK_ID ENFERMEDAD	Primary_Key	(null)	(null)	(null)	(null)
2 SYS_C005078	Check	"ID ENFERMEDAD" IS NOT NULL	(null)	(null)	(null)
3 UK_ENF_VARIANTE_NOMBRE	Unique	(null)	(null)	(null)	(null)

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_OWNER	R_TABLE_NAME	R_CONSTRAINT_NAME
1 PK_ID GUACHERA	Primary_Key	(null)	(null)	(null)	(null)
2 SYS_C005081	Check	"ID GUACHERA" IS NOT NULL	(null)	(null)	(null)

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_OWNER	R_TABLE_NAME	R_CONSTRAINT_NAME
1 PK_ID MEDICAMENTO	Primary_Key	(null)	(null)	(null)	(null)
2 SYS_C005085	Check	"ID MEDICAMENTO" IS NOT NULL	(null)	(null)	(null)
3 SYS_C005086	Check	"NRO_MEDICAMENTO" IS NOT NULL	(null)	(null)	(null)
4 UK_NRO_MEDICAMENTO	Unique	(null)	(null)	(null)	(null)

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_OWNER	R_TABLE_NAME	R_CONSTRAINT_NAME
1 FK_ID_ENFERMEDAD PADECE	Foreign_Key	(null)	PDT	ENFERMEDAD	PK_ID_ENFERMEDAD
2 FK_ID_TERNERA PADECE	Foreign_Key	(null)	PDT	TERNERA	PK_ID_TERNERA
3 PK_ID PADECE	Primary_Key	(null)	(null)	(null)	(null)
4 SYS_C005098	Check	"ID PADECE" IS NOT NULL	(null)	(null)	(null)

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_OWNER	R_TABLE_NAME	R_CONSTRAINT_NAME
1 FK_TERNERA_HIST_PESO	Foreign_Key	(null)	PDT	TERNERA	UK_CARAVANA
2 PK_ID REGISTRO_TP	Primary_Key	(null)	(null)	(null)	(null)
3 SYS_C005110	Check	"ID REGISTRO_TP" IS NOT NULL	(null)	(null)	(null)
4 SYS_C005111	Check	"PESO" IS NOT NULL	(null)	(null)	(null)
5 SYS_C005112	Check	"FECHA" IS NOT NULL	(null)	(null)	(null)
6 SYS_C005113	Check	"CARAVANA_SNIG" IS NOT NULL	(null)	(null)	(null)
7 SYS_C005114	Check	"NOMBRE_USUARIO" IS NOT NULL	(null)	(null)	(null)

CONSTRAINT_NAME	CONSTRAINT_TYPE	SEARCH_CONDITION	R_OWNER	R_TABLE_NAME	R_CONSTRAINT_NAME
1 FK_TERNERA_TT	Foreign_Key	(null)	PDT	TERNERA	UK_CARAVANA
2 PK_ID REGISTRO_TT	Primary_Key	(null)	(null)	(null)	(null)
3 SYS_C005118	Check	"ID REGISTRO_TT" IS NOT NULL	(null)	(null)	(null)
4 SYS_C005119	Check	"TEMPERATURA" IS NOT NULL	(null)	(null)	(null)
5 SYS_C005120	Check	"FECHA" IS NOT NULL	(null)	(null)	(null)
6 SYS_C005121	Check	"CARAVANA_SNIG" IS NOT NULL	(null)	(null)	(null)
7 SYS_C005122	Check	"NOMBRE_USUARIO" IS NOT NULL	(null)	(null)	(null)



❖ CONSTRAINT_NAME	❖ CONSTRAINT_TYPE	SEARCH_CONDITION	❖ R_OWNER	❖ R_TABLE_NAME	❖ R_CONSTRAINT_NAME
1 FK_ID_MEDICAMENTO_SUMINISTRA	Foreign_Key	(null)	PDT	MEDICAMENTO	PK_ID_MEDICAMENTO
2 FK_ID_TERNERA_SUMINISTRA	Foreign_Key	(null)	PDT	TERNERA	PK_ID_TERNERA
3 PK_ID_SUMINISTRA	Primary_Key	(null)	(null)	(null)	(null)
4 SYS_C005102	Check	"ID_SUMINISTRA" IS NOT NULL	(null)	(null)	(null)

❖ CONSTRAINT_NAME	❖ CONSTRAINT_TYPE	SEARCH_CONDITION	❖ R_OWNER	❖ R_TABLE_NAME	❖ R_CONSTRAINT_NAME
1 FK_ID_TERNERA_GUACHERA	Foreign_Key	(null)	PDT	GUACHERA	PK_ID_GUACHERA
2 PK_ID_TERNERA	Primary_Key	(null)	(null)	(null)	(null)
3 SYS_C005089	Check	"ID_TERNERA" IS NOT NULL	(null)	(null)	(null)
4 SYS_C005090	Check	"ID_GUACHERA" IS NOT NULL	(null)	(null)	(null)
5 UK_CARAVANA	Unique	(null)	(null)	(null)	(null)

❖ CONSTRAINT_NAME	❖ CONSTRAINT_TYPE	SEARCH_CONDITION	❖ R_OWNER	❖ R_TABLE_NAME	❖ R_CONSTRAINT_NAME
1 PK_ID_USUARIO	Primary_Key	(null)	(null)	(null)	(null)
2 SYS_C005126	Check	"ID_USUARIO" IS NOT NULL	(null)	(null)	(null)
3 SYS_C005127	Check	"NOMBRE" IS NOT NULL	(null)	(null)	(null)
4 SYS_C005128	Check	"APELLIDO" IS NOT NULL	(null)	(null)	(null)
5 SYS_C005129	Check	"NOMBRE_USUARIO" IS NOT NULL	(null)	(null)	(null)
6 SYS_C005130	Check	"CONTRASENIA" IS NOT NULL	(null)	(null)	(null)
7 UK_NOMBRE_USUARIO	Unique	(null)	(null)	(null)	(null)

❖ CONSTRAINT_NAME	❖ CONSTRAINT_TYPE	SEARCH_CONDITION	❖ R_OWNER	❖ R_TABLE_NAME	❖ R_CONSTRAINT_NAME
1 FK_ID_GUACHERA_VIVE	Foreign_Key	(null)	PDT	GUACHERA	PK_ID_GUACHERA
2 FK_TERNERA_VIVE	Foreign_Key	(null)	PDT	TERNERA	PK_ID_TERNERA
3 PK_ID_VIVE	Primary_Key	(null)	(null)	(null)	(null)
4 SYS_C005094	Check	"ID_VIVE" IS NOT NULL	(null)	(null)	(null)

BASE DE DATOS CORPORATIVA

1. Introducción

1.1 ¿Qué es?

Esté permite el almacenamiento y manejo de datos organizados de una forma activa y fácil. Además, permite el análisis de datos.

1.2 ¿Por qué es útil la BDC?

Usar BDC puede traer los siguientes beneficios:

- Capacidad de almacenar y controlar una gran cantidad de datos de manera eficiente
- Se puede permitir manejar datos en por distintos medios de manera segura
- Acceder a los datos de manera rápida y fácil
- Se diseña a la realidad que uno tenga

2. Tablas

2.1 Definición

Las tablas se puede dividir entre dos tipos, tabla de hechos y tabla de dimensiones.

- **Tabla de hechos:** este tiene el componente principal de los datos, el objeto principal, donde están los campos de datos principales.
- **Tabla de dimensiones:** esta contiene las medidas, las diferentes características o grupos que se quiere medir.

Estas dos tipos de tablas están relacionadas por el hecho de que la tabla de hechos tiene todos los atributos de las tablas de dimensiones, y estas tablas almacenan estos datos de una manera más específica.

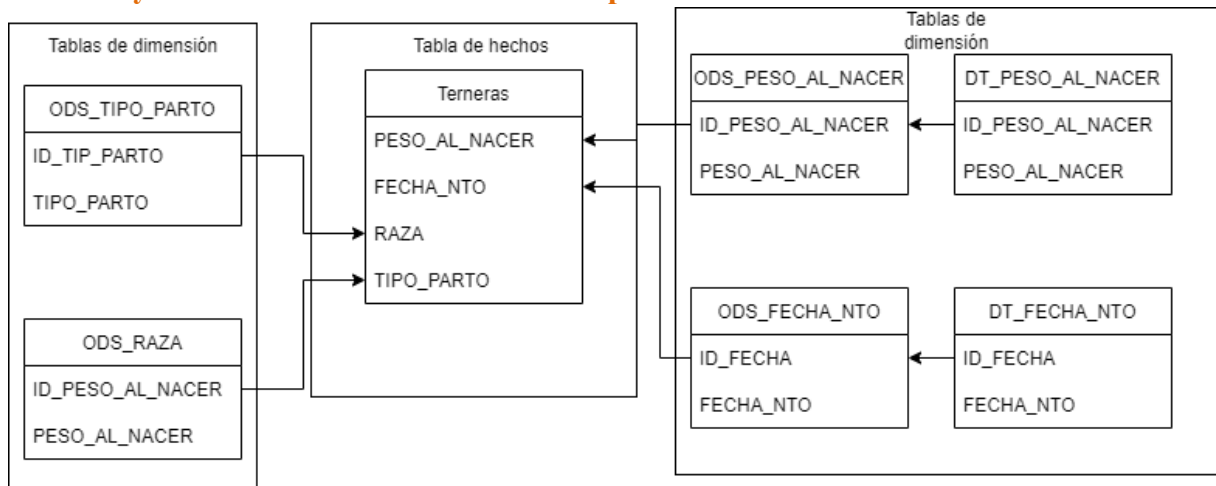
2.1 Tabla de hechos

Nuestra tabla de hechos es la tabla **Terneras**, porque es nuestro principal objeto de interés, además de estar relacionado con todo el operativo. Esta contiene los datos más importantes que se quiere saber de la ternera que son, su peso al nacer, la fecha en que nació, su tipo de raza y el tipo de parto que tuvo.

2.2 Tabla de dimensión

Estas tablas contienen los datos de manera separada y organizada de la ternera. Los atributos de la tabla ternera, están mejor dimensionadas en estas tablas.

3. Tablas y relaciones de la base de datos corporativa.



DIMENSION	TABLA	PRIMARY KEY	DATO
	DT_FECHA_NTO	ID_FECHA	FECHA_NTO
	ODS_FECHA_NTO	ID_FECHA	FECHA_NTO
	DT_PESO_AL_NACER	ID_PESO_AL_NACER	PESO_AL_NACER
	ODS_PESO_AL_NACER	ID_PESO_AL_NACER	PESO_AL_NACER
	ODS_RAZA	ID_RAZA	RAZA
	ODS_TIPO_PARTO	ID_TIPO_PARTO	TIPO_PARTO

HECHOS	TABLA	Columna	TIPO ATRIBUTO
	HT_TERNERAS	SK_FECHA_NTO	DIMENSIÓN
		SK_PESO_AL_NACER	DIMENSIÓN
		SK_RAZA	DIMENSIÓN
		SK_TIPO_PARTO	DIMENSIÓN

Usuario

El DWH (data warehouse), estará funcionando en un esquema diferente al de la BD Operativa, debido a que de esta manera se realiza en la realidad, BD operativa por un lado y BDC por otro. Para esto, creamos un usuario BDC_THEBOYS con el usuario Administrador (SYSTEM) y la siguiente SQL query:

```
CREATE USER BDC_THEBOYS IDENTIFIED BY BDC_THEBOYS DEFAULT
TABLESPACE USERS;
```

Posteriormente, le asignamos permisos de Administrador, para que cree y gestione la BDC.

GRANT ALL PRIVILEGES TO BDC_THEBOYS;

Creación de todas las tablas correspondientes al modelo:

Según lo hablado en la entrega del primer sprint se identificarlos y crearon las siguientes tablas de dimensiones: DT_FECHA_NTO, DT_RAZA, DT_TIPO_DE_PARTO, DT_PESO_AL_NACER

Creación de tabla de hechos del modelo:

Se solicita que la tabla de hechos del modelo dimensional sea enfocado a las terneras, por lo cual en la BDC creamos la tabla de hechos HT_TERNERAS.

Creación de secuencias y triggers:

Para insertar datos en las tablas de dimensiones, creamos secuencias y triggers para cada una de las tablas, las cuales nos va a asignar un ID cada vez que se inserte un dato nuevo en las tablas.

```
CREATE SEQUENCE SEQ_TIPO_PARTO
START WITH 1
INCREMENT BY 1;

CREATE SEQUENCE SEQ_RAZA
START WITH 1
INCREMENT BY 1;

CREATE SEQUENCE SEQ_PESO_AL_NACER
START WITH 1
INCREMENT BY 1;

CREATE SEQUENCE SEQ_FECHA_NTO
START WITH 1
INCREMENT BY 1;
```

```
CREATE OR REPLACE TRIGGER T_TERNERAS_SEQ
BEFORE INSERT
ON ODS_TIPO_PARTO
FOR EACH ROW
BEGIN
    SELECT SEQ_TIPO_PARTO.NEXTVAL
    INTO :NEW.ID_TIP_PARTO
    FROM DUAL;
END T_TERNERAS_SEQ;
```

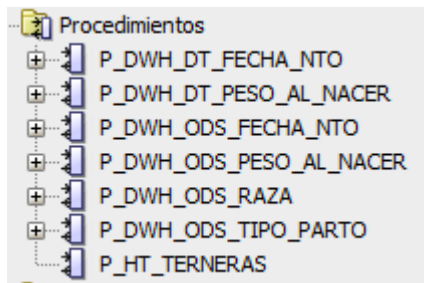
```
CREATE OR REPLACE TRIGGER T_RAZA_SEQ
BEFORE INSERT
ON ODS_RAZA
FOR EACH ROW
BEGIN
    SELECT SEQ_RAZA.NEXTVAL
    INTO :NEW.ID_RAZA
    FROM DUAL;
END T_RAZA_SEQ;

CREATE OR REPLACE TRIGGER T_PESO_AL_NACER_SEQ
BEFORE INSERT
ON ODS_PESO_AL_NACER
FOR EACH ROW
BEGIN
    SELECT SEQ_PESO_AL_NACER.NEXTVAL
    INTO :NEW.ID_PESO_AL_NACER
    FROM DUAL;
END T_PESO_AL_NACER_SEQ;

CREATE OR REPLACE TRIGGER T_FECHA_NTO_SEQ
BEFORE INSERT
ON ODS_FECHA_NTO
FOR EACH ROW
BEGIN
    SELECT SEQ_FECHA_NTO.NEXTVAL
    INTO :NEW.ID_FECHA
    FROM DUAL;
END T_FECHA_NTO_SEQ;
```


Creación de procedimiento para cargar datos

Se crearán procedimiento para cargar los datos en cada una de las tablas de dimensiones



```
create or replace NONEDITIONABLE PROCEDURE P_DWH_DT_FECHA_NTO
IS
BEGIN
MERGE INTO DT_FECHA_NTO a
USING
(SELECT DISTINCT FECHA_NTO FROM DT_TERNERAS WHERE FECHA_NTO IS NOT NULL) b
ON
(a.FECHA_NTO = b.FECHA_NTO)
WHEN NOT MATCHED THEN INSERT (FECHA_NTO)
VALUES
(F_ESTACIONES(b.FECHA_NTO))
;
COMMIT;
END P_DWH_DT_FECHA_NTO;
```

```
create or replace NONEDITIONABLE PROCEDURE P_DWH_DT_PESO_AL_NACER
IS
BEGIN
MERGE INTO DT_PESO_AL_NACER a
USING
(SELECT DISTINCT PESO_AL_NACER FROM DT_TERNERAS WHERE PESO_AL_NACER IS NOT NULL) b
ON
(a.PESO_AL_NACER = b.PESO_AL_NACER)
WHEN NOT MATCHED THEN INSERT (PESO_AL_NACER)
VALUES
(F_RANGO_PESO(b.PESO_AL_NACER))
;
COMMIT;
END P_DWH_DT_PESO_AL_NACER;
```



```
create or replace NONEDITIONABLE PROCEDURE P_DWH_ODS_FECHA_NTO
IS
BEGIN
MERGE INTO ODS_FECHA_NTO a
USING
(SELECT DISTINCT FECHA_NTO FROM DT_FECHA_NTO WHERE FECHA_NTO IS NOT NULL) b
ON
(a.FECHA_NTO = b.FECHA_NTO)
WHEN NOT MATCHED THEN INSERT (FECHA_NTO)
VALUES
(b.FECHA_NTO)
;
COMMIT;
END P_DWH_ODS_FECHA_NTO;
```

```
create or replace NONEDITIONABLE PROCEDURE P_DWH_ODS_PESO_AL_NACER
IS
BEGIN
MERGE INTO ODS_PESO_AL_NACER a
USING
(SELECT DISTINCT PESO_AL_NACER FROM DT_PESO_AL_NACER WHERE PESO_AL_NACER IS NOT NULL) b
ON
(a.PESO_AL_NACER = b.PESO_AL_NACER)
WHEN NOT MATCHED THEN INSERT (PESO_AL_NACER)
VALUES
(b.PESO_AL_NACER)
;
COMMIT;
END P_DWH_ODS_PESO_AL_NACER;
```

```
create or replace NONEDITIONABLE PROCEDURE P_DWH_ODS_RAZA
IS
BEGIN
MERGE INTO ODS_RAZA a
USING
(SELECT DISTINCT RAZA FROM DT_TERNERAS WHERE RAZA IS NOT NULL ) b
ON
(a.DSC_RAZA = b.RAZA)
WHEN NOT MATCHED THEN INSERT (DSC_RAZA)
VALUES
(b.RAZA)
;
COMMIT;
END P_DWH_ODS_RAZA;
```

```
create or replace NONEDITIONABLE PROCEDURE P_DWH_ODS_TIPO_PARTO
IS
BEGIN
MERGE INTO ODS_TIPO_PARTO a
USING
(SELECT DISTINCT TIPO_PARTO FROM DT_TERNERAS WHERE TIPO_PARTO IS NOT NULL) b
ON
(a.TIPO_PARTO = b.TIPO_PARTO)
WHEN NOT MATCHED THEN INSERT (TIPO_PARTO)
VALUES
(b.TIPO_PARTO)
;
COMMIT;
END P_DWH_ODS_TIPO_PARTO;
```

Creación de función para estaciones

Para la carga de las fechas, no ingresamos todas las fechas en si, si no que agregamos para cada rango de fechas unas estación, en la tabla de dimensiones vamos a ingresarlo ejecutando el siguiente procedimiento

```
CREATE OR REPLACE FUNCTION F_ESTACIONES (FECHA DATE)
RETURN VARCHAR
IS
BEGIN
RETURN
(
CASE
WHEN TO_CHAR(FECHA, 'MM') >='03' AND TO_CHAR(FECHA, 'MM') <='06' THEN
CASE
WHEN TO_CHAR(FECHA, 'DD')>'22'
THEN 'OTOÑO'
ELSE 'VERANO'
END
WHEN TO_CHAR(FECHA, 'MM') >='06' AND TO_CHAR(FECHA, 'MM') <='09' THEN
CASE
WHEN TO_CHAR(FECHA, 'DD')>'22'
THEN 'INVIERNO'
ELSE 'OTOÑO'
END
WHEN TO_CHAR(FECHA, 'MM') >='09' AND TO_CHAR(FECHA, 'MM') <='12' THEN
CASE
WHEN TO_CHAR(FECHA, 'DD')>'22'
THEN 'PRIMAVERA'
ELSE 'INVIERNO'
END
WHEN TO_CHAR(FECHA, 'MM') >='12' AND TO_CHAR(FECHA, 'MM') <='06' THEN
CASE
CASE
WHEN TO_CHAR(FECHA, 'DD')>'22'
THEN 'VERANO'
ELSE 'PRIMAVERA'
END
END
);
END F_ESTACIONES;
```

Creación de función para peso al nacer

En esta función cuando una ternera se encuentra en determinado rango de peso nos va a devolver si es un peso BAJO,PROMEDIO,ALTO.

```
CREATE OR REPLACE FUNCTION F_RANGO_PESO (PESO NUMBER)
RETURN VARCHAR
IS
BEGIN
RETURN
(
CASE
WHEN PESO<30
    THEN 'BAJO'
WHEN PESO>30 and PESO<32
    THEN 'PROMEDIO'
WHEN PESO>32
    THEN 'ALTO'
END
);
END F_RANGO_PESO;
```

Carga de datos en tabla de hechos de terneras

Como último se desarrolló un procedimiento para la carga de la tabla de hechos en la DWH, una vez cargadas todas las dimensiones, lo ejecutamos

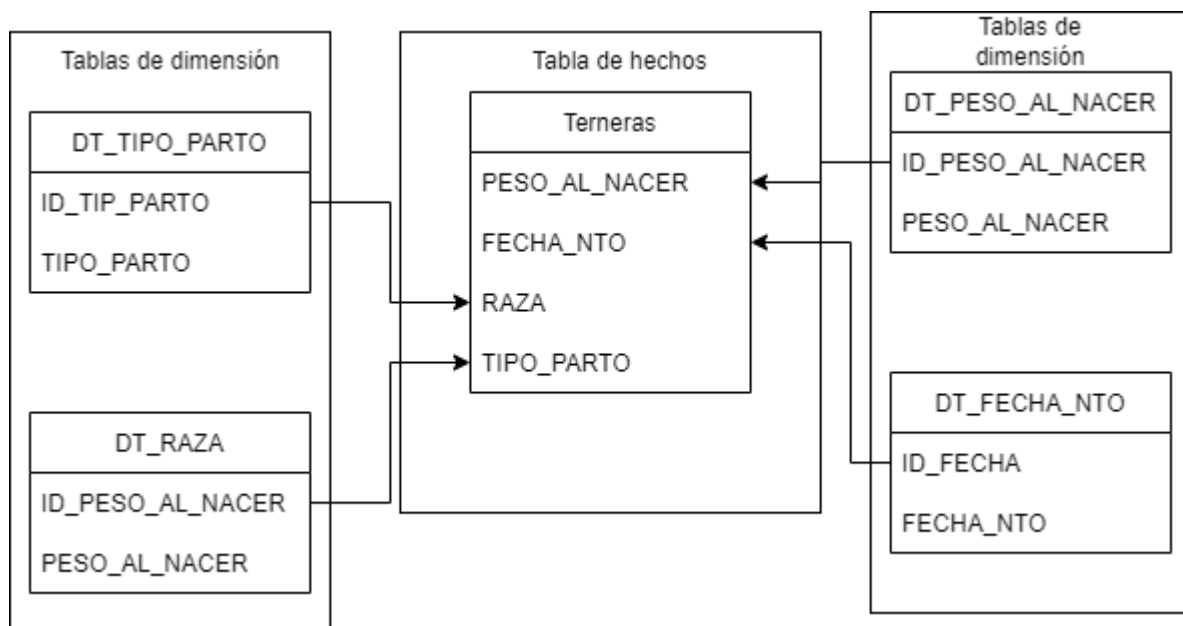
```
create or replace NONEDITIONABLE PROCEDURE P_HT_TERNERAS
IS
BEGIN
DELETE FROM HT_TERNERAS;
INSERT INTO HT_TERNERAS
(FECHA_NTO, PESO_AL_NACER, RAZA, TIPO_PARTO)
SELECT
AA.ID_FECHA,
BB.ID_PESO_AL_NACER,
CC.ID_RAZA,
DD.ID_TIP_PARTO
FROM DT_TERNERAS T
LEFT OUTER JOIN ODS_FECHA_NTO AA
ON AA.FECHA_NTO = F_ESTACIONES(T.FECHA_NTO)
LEFT OUTER JOIN ODS_PESO_AL_NACER BB
ON BB.PESO_AL_NACER = F_RANGO_PESO(T.PESO_AL_NACER)
LEFT OUTER JOIN ODS_RAZA CC
ON CC.DSC_RAZA = T.RAZA
LEFT OUTER JOIN ODS_TIPO_PARTO DD
ON DD.TIPO_PARTO = T.TIPO_PARTO
;
END P_HT_TERNERAS;
```

Cambios para Sprint 3:

Se cambiaron el nombre de las tablas para que correspondan con las tablas que son, Las ODS van a ser tablas en las cuales se van a insertar los datos un momento para arreglar los datos ingresados desde la tabla de terneras en este caso. y las otras tablas de hechos(DT) como tal.

Además de esto se ajustaron todos los procedimientos, triggers, etc. para que correspondan con el nombre de las tablas nuevamente.

Además de esto se ajustó el esquema en donde se eliminaron las tablas ODS y se agregaron nuevos campos para llevar a cabo las métricas que serán representadas con la utilización del software Power Bi.



Tablas creadas en la DWH

DIMENSIÓN	TABLA	PRIMARY KEY	DATO
	ODS_FECHA_NTO	ID_FECHA	FECHA_NTO
	DT_FECHA_NTO	ID_FECHA	FECHA_NTO
	ODS_PESO_AL_NACER	ID_PESO_AL_NACER	PESO_AL_NACER
	DT_PESO_AL_NACER	ID_PESO_AL_NACER	PESO_AL_NACER
	DT_RAZA	ID_RAZA	RAZA
	DT_TIPO_PARTO	ID_TIPO_PARTO	TIPO_PARTO

HECHOS	TABLA	Columna	TIPO ATRIBUTO
	HT_TERNERAS	SK_FECHA_NTO	DIMENSIÓN
		SK_PESO_AL_NACER	DIMENSIÓN
		SK_RAZA	DIMENSIÓN
		SK_TIPO_PARTO	DIMENSIÓN
		CONTADOR	MÉTRICA
		PESO	MÉTRICA
		FECHA_NACIMIENTO	MÉTRICA

PROCESO ETL

1. Introducción

1.1 ¿Qué es ETL?

Es una característica propia de las bases de datos que es el cómo se gestionan los datos de esta base de datos, esto por medio de una aplicación que se conecta a la base de datos corporativa.

2. Proceso ETL

Para ingresar a los datos, se realiza por medio de una aplicación web y una aplicación móvil; estas permiten el ingreso, la carga y la modificación de los datos de la base de datos. Pero no todos pueden manejar los datos como quiera, hay una tabla que almacena los usuarios y cada uno tiene sus respectivos permisos, que como consecuencia cada usuario modificará los datos de distintas maneras.

Se puede ingresar y modificar los datos relacionados con las terneras, al momento de ingresarla a la base de datos, que pasa al momento de nacer la ternera. Las terneras también serán relacionadas con otros datos importantes, como pueden ser las enfermedades que tiene, los medicamentos que se le proporciona, donde será alojado, y demás información.

Además, según el rol del usuario, puede modificar los datos, cuando sea necesario; ejemplo, una ternera tenga una enfermedad, ese dato será ingresado con que enfermedad padece y cuando; pero cuando se cure de la enfermedad que tenía, ahí se modificará el dato de la ternera de enfermo a sin enfermedad.

También se puede eliminar datos si se considera necesario, claro está que según el rol porque no todos pueden hacerlo, al igual que la modificación.

Power BI

En esta sección de habla de los detalles del Power BI

1. Introducción

1.1 ¿Qué es Power BI?

Power BI es una herramienta para analizar datos en forma de gráficos personalizados; con interfaz simple y rápida.

1.2 Beneficios

- *Generar gráficos*
 - Simples
 - Rápidos
 - Personalizables
 - Generados según nuestras necesidades
 - Diferentes modelos dependiendo de lo que se quiera analizar
- *Datos*
 - A simple vista
 - Facilidad de lectura
 - Mejora en la eficiencia del análisis de datos

2. Power BI

Nuestro Power BI consiste en la visualización de los datos relacionados con las terneras. Estos datos estarán previamente cargados por medio del uso de la aplicación.

2.1 Pestañas

Está formada por 8 pestañas en total:

1. **Inicio:** pestaña que sirve para navegar entre las otras pestañas de gráficos
2. **Cantidad:** pestaña donde se ve datos relacionados con la cantidad de terneras que nacieron por año, estación y meses.
3. **Costos:** pestaña donde están los datos relacionados con los costos que genera la ternera en temas de alimentación y medicamentos, esto organizados por sus respectivas maracas y tipos
4. **Salud:** pestaña donde se ve los datos relacionados con la salud de las terneras, consistiendo en si están sanos o no de las distintas enfermedades que padecieron o padecen; junto a un gráfico que muestra los medicamentos usados en cada enfermedad, con un filtro por marcas
5. **Peso:** pestaña donde se ve los datos de peso al nacer, y la estándar de las terneras, organizadas por su índice correspondiente y por las razas. Además de un filtro de alimento proporcionado.
6. **Nacimientos:** pestaña de datos que muestra la cantidad de partos naturales y distócicos (que necesitaron ayuda externa).
7. **Vive:** pestaña donde se muestra la cantidad de terneras que viven por zonas y guacheras.
8. **Temperatura:** donde se ve el índice de temperatura según la estación del año y la raza.

2.2 Filtros

En total son 5 filtros

1. Quitar filtros

- a. **Uso:** sirve para quitar los filtros puestos, y dejar en su manera original.
- b. **Ubicación:** todas las pestañas con gráficos

2. Filtros por año

- a. **Uso:** sirve para filtrar los datos por año
- b. **Ubicación:** Todas las pestañas de gráficos

3. Filtros por estación

- a. **Uso:** sirve para filtrar los datos según la estación (verano, otoño, invierno y primavera)
- b. **Ubicación:** Costos, Salud, Peso y Vive

4. Filtro por marca de medicamento

- a. **Uso:** filtra los datos según la marca de los medicamentos utilizados
- b. **Ubicación:** pestaña Salud

5. Filtro por alimentos consumidos

- a. **Uso:** filtra los datos según los alimentos proporcionados a las terneras
- b. **Ubicación:** pestaña Peso

Además, se puede agregar un extra de filtro seleccionando alguna de las categorías u objetos visuales de las gráficas. Por ejemplo, en la pestaña cantidad puedo seleccionar una de las razas y en las gráficas se resaltarán los tipos de datos seleccionados.

2.3 Visual

En el apartado visual se optó por un esquema de colores azules, naranjas, negros, grises y blancos. En concreto, blanco y gris para fondos, negro para letras, y azules y naranjas para objetos visuales y botones.

Los botones son tienen como color de fondo blanco con letras o iconos negros y los bordes son de color azul oscuro y naranja; además de que al posicionar el cursor encima de estos botones se puede ver una pequeña descripción de su función.

Las gráficas tienen como color de fondo blanco con letras de color negro; y los objetos visuales son con la variación de colores naranjas y azules en distintos tonos; también tienen pequeños números que muestran la cantidad que se tiene en los datos, para así tener una mejor visualización de los datos.

Para cada pestaña se tiene una etiqueta visual con su icono asignado y su nombre para tener una mejor orientación al navegar por las diferentes pestañas; además de que estas etiquetas coinciden con los botones que se pueden utilizar para navegar entre páginas en la pestaña inicio.

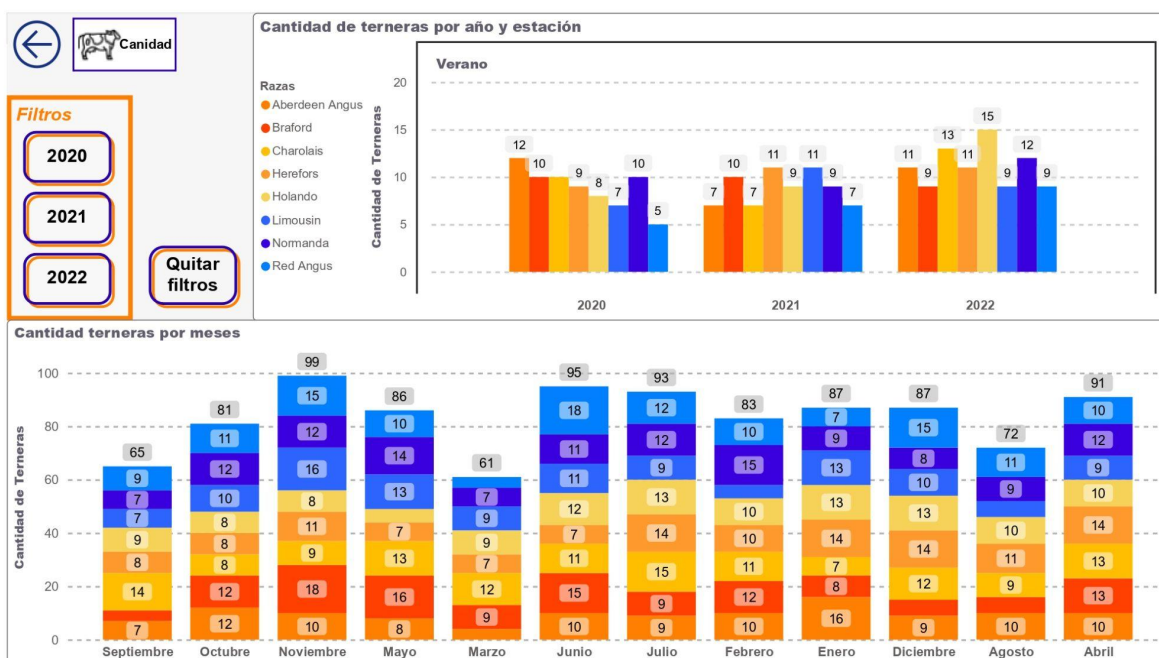
2.4 Imágenes de las pestañas

Ahora en esta sección se muestra como se verá las pestañas y gráficas.

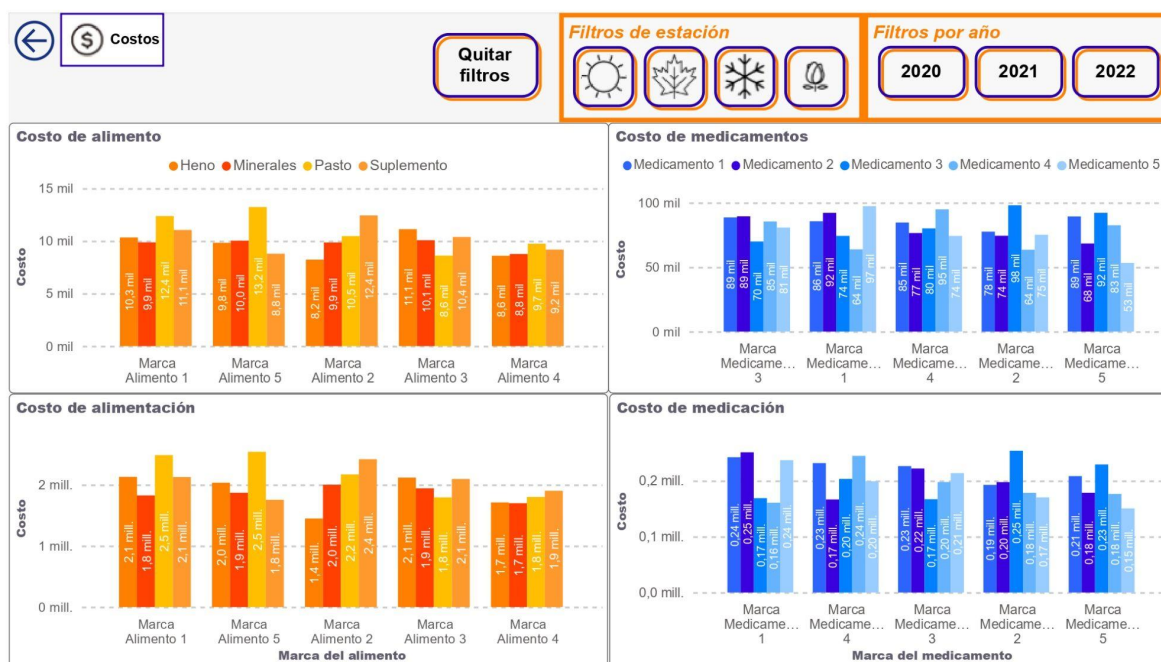
Inicio



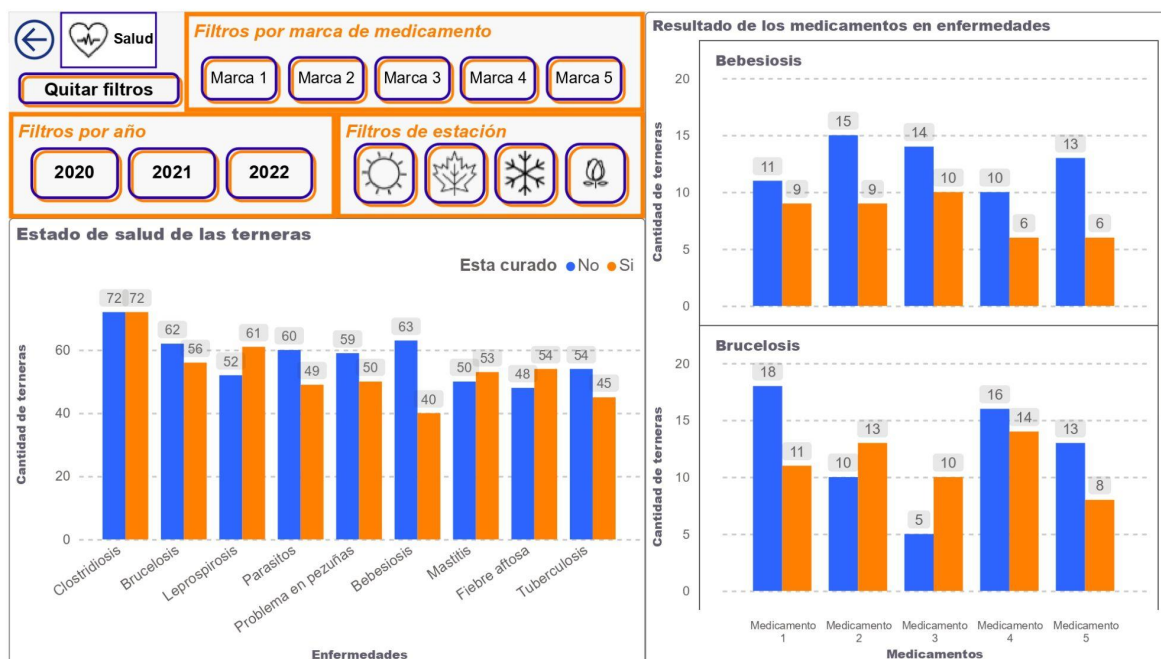
Cantidad



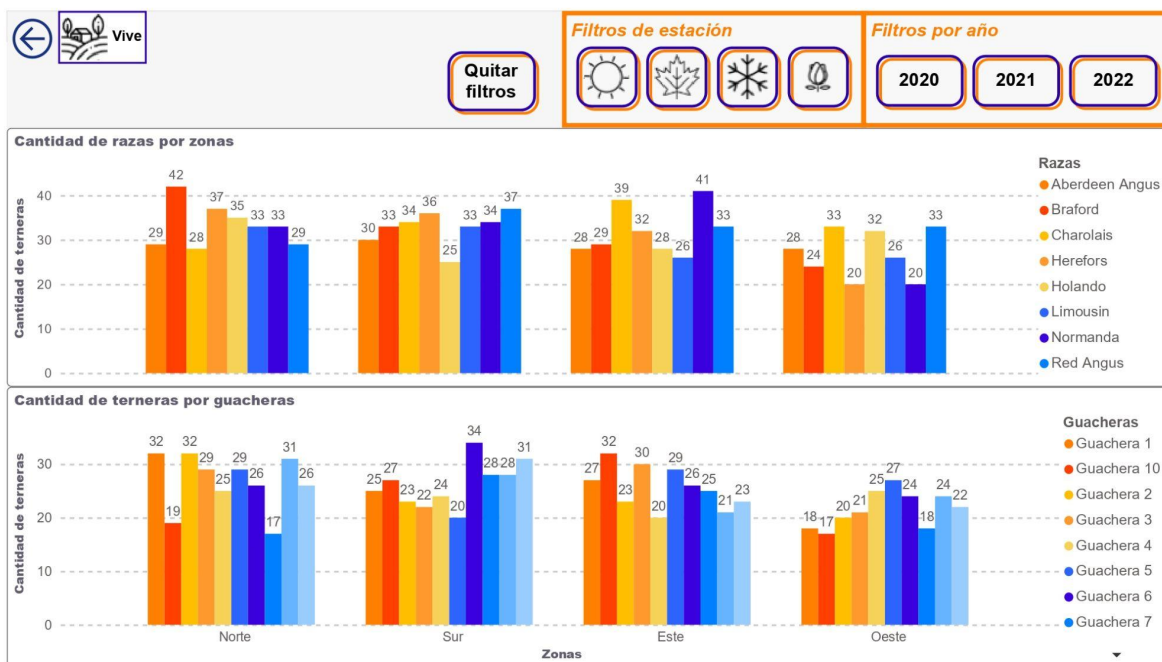
Costos



Salud



Vive



Temperatura

