

# Ejercicio Full Stack Developer - Bastó

En Bastó nos encargamos de digitalizar el campo y armar una nueva plataforma para la ganadería del futuro. Es por ello y como parte de este trabajo que será necesario que digitalizemos los animales de un campo.

Este ejercicio consta de la creación de un formulario CRUD (Altas, Bajas, Modificaciones) de datos de vacas.

Los datos de los animales a guardar y sus tipos son:

- ID SENASA (Alfanumerico 16 chars) - Requerido
- Tipo Animal (Novillo, Toro, Vaquillona) - Requerido
- Peso animal (kg - numerico) - Opcional
- Nombre de potrero.(Texto hasta 200 chars) - Requerido
- Tipo de Dispositivo (COLLAR, CARAVANA) - Requerido
- Número de dispositivo. (Alfanumerico 8 chars) - Requerido

Ejemplos de interfaces gráficas (son a modo de ejemplo de flujo no tienen que ser exactamente iguales):

Basto

Menu / Animales

### Gestion de animales

[Nuevo Animal](#)

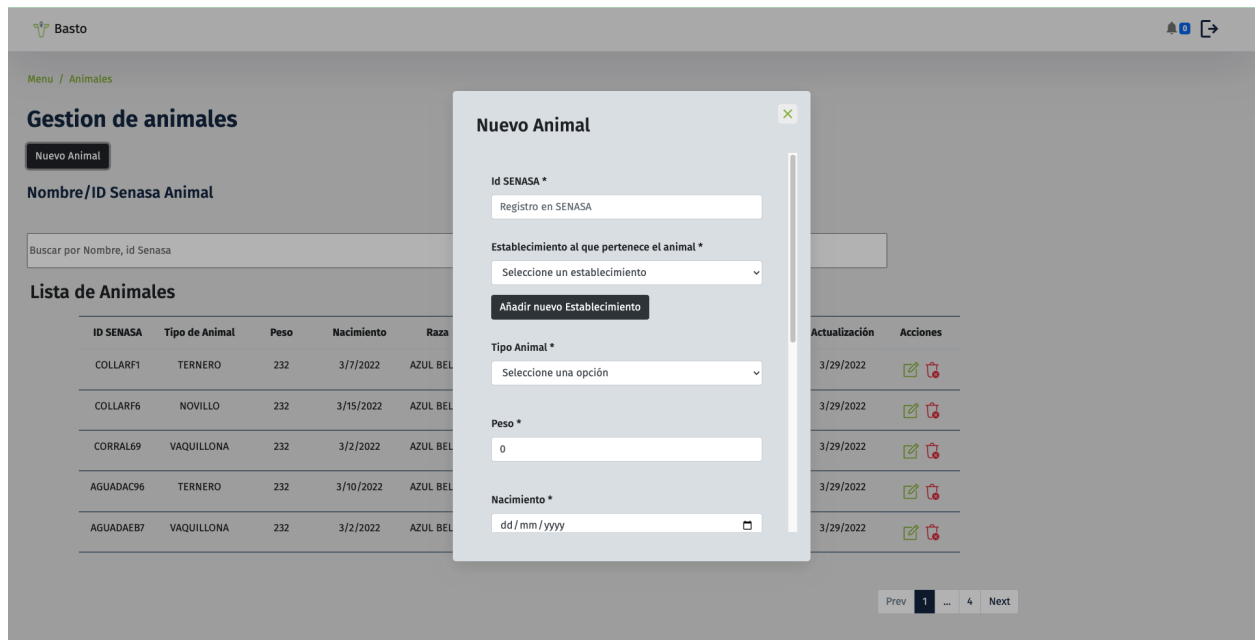
Nombre/ID Senasa Animal

Buscar por Nombre, id Senasa

#### Lista de Animales

ID SENASA	Tipo de Animal	Peso	Nacimiento	Raza	Preñada	Fecha de Parto	Observaciones	Creación	Actualización	Acciones
COLLARF1	TERNERO	232	3/7/2022	AZUL BELGA	NO	SIN DATOS		3/29/2022	3/29/2022	<a href="#">Editar</a> <a href="#">Eliminar</a>
COLLARF6	NOVILLO	232	3/15/2022	AZUL BELGA	NO	SIN DATOS		3/29/2022	3/29/2022	<a href="#">Editar</a> <a href="#">Eliminar</a>
CORRAL69	VAQUILLONA	232	3/2/2022	AZUL BELGA	NO	SIN DATOS		3/29/2022	3/29/2022	<a href="#">Editar</a> <a href="#">Eliminar</a>
AGUADAC96	TERNERO	232	3/10/2022	AZUL BELGA	NO	SIN DATOS		3/29/2022	3/29/2022	<a href="#">Editar</a> <a href="#">Eliminar</a>
AGUADAE7	VAQUILLONA	232	3/2/2022	AZUL BELGA	NO	SIN DATOS		3/29/2022	3/29/2022	<a href="#">Editar</a> <a href="#">Eliminar</a>

Prev 1 ... 4 Next



### Aclaraciones sobre el ejercicio:

- Se puede utilizar librerías externas y frameworks de UI con criterio.
- Alta, eliminación, y modificación deben funcionar correctamente.
- Solo debe funcionar en desktop (No importa mobile)
- Se debe desarrollar el frontend y backend REST. (React.js y node.js con Express)
- La barra superior con menú es opcional.
- No tiene que soportar login de usuarios.

### Que se va a evaluar:

- Calidad del código de React.js y node.js/Express. Arquitectura utilizada.
- Código/comentarios del código escrito en inglés. (No se corrige código en español)
- Flujo y funcionamiento del formulario. (Bugs)
- Tests unitarios de Frontend y Backend incluidos.
- Tiempo de entrega (se estima 1 semana desde el envío)

### Tecnologías a utilizar (OBLIGATORIO):

- React.js
- MongoDB
- Express/node.js

### Entregables (OBLIGATORIO):

- Link al repositorio de Github con el código.
- El README.md del proyecto tiene que tener todas las instrucciones correctas para probarlo localmente.

Dudas o consultas sobre el ejercicio se pueden hacer al Líder Técnico de Software:

Ing. Emanuel Ravera ([ravera.emanuel@gmail.com](mailto:ravera.emanuel@gmail.com))