**Gestión Operativa del Proceso de Crianza de las Terneras de Tambo**

Equipo:

**Ti-Rex Software Factory**

Estudiantes:

* Noel Charrute
* Martin Cáceres
* Camilo Ferreira
* Gonzalo Alfonso

Tutores:

* Alejandra Viglietti
* Leonardo Pera
* Fernando Montañes
* Natalia Botto

RESUMEN

Para el desarrollo de este proyecto continuamos con lo trabajado en el “Caso de estudio” del primer semestre, donde nos planteamos resolver el problema planteado por la Carrera de Producción Lechera sobre la crianza de terneras de tambo, ya que la realidad planteada es que hoy en día no se cuenta con un software específico que atienda las necesidades de tamberos de tener un control de sus guacheras por medio de informes y análisis de determinados eventos que pueden ayudar a organizar planes para el mejor aprovechamiento de sus recursos y también a no perder vidas de terneros por no tener reportes de su estado de salud. Por parte del equipo de desarrollo se buscó atender todas las necesidades planteadas e implementar una aplicación con una interfaz amigable y de fácil uso para el personal del tambo, además de eso es totalmente escalable para futuras implementaciones de nuevas funcionalidades. En este documento de proyecto se desarrollan las metodologías que se utilizaron, el proceso de desarrollo de la solución, conclusiones y recomendaciones para continuar con el mismo.

ABSTRACT

For the development of this project we continue with the work in the "Case Study" of the first semester, where we set out to solve the problem posed by the Dairy Production college career on the raising of the tambo lands, that the reality raised is that today it is day there is no specific software that meets the needs of dairy farmers to have control of their accounts through reports and analysis of events that can help plan for the best use of their resources and also not to lose calf lives for not having reports of his health status. On the part of the development team you can use all the needs and implement an application with a friendly and user-friendly interface for the staff of the dairy, besides that it is fully scalable for future implementations of new functionalities. In this project document are made the methodologies used, the process of developing the solution, conclusions and recommendations to continue with it.

**Contenido**

[Introducción 3](#_30j0zll)

[Marco Teórico 3](#_1fob9te)

[Marco Metodológico 3](#_3znysh7)

[Proceso realizado y Resultados 3](#_2et92p0)

[Conclusiones 3](#_tyjcwt)

[Recomendaciones 4](#_3dy6vkm)

[Anexos 4](#_1t3h5sf)

## Introducción

Se desarrollará una aplicación la cual pueda brindar herramientas de gestión integral para tambos. Entre factores a destacar de la gestión podemos mencionar administración de inventario de alimentos, registro de consumo de alimentos, administración de inventario de medicamentos, seguimiento de desarrollo de terneros, entre otros.

* 1. **Propósito**

El propósito de este proyecto es solucionar el problema de crianza de terneras de tambo por medio de un software diseñado para atender las necesidades de tamberos.

* 1. **Alcance**

En este caso el alcance es definido en base a los requerimientos asignados al equipo de parte de los tutores. Por lo tanto, esta aplicación contempla las siguientes funcionalidades:

* Costo de alimentación por Guachera
* Registro de consumo de alimento
* Análisis de eventos clínicos
* Alta de alimentos
* ABM de Usuarios

## Marco Teórico

* Base de datos
  + Diseño y modelado físico de la base datos
    - Se empleó MER como punto de partida inicial para analizar la realidad específica a resolver
  + Se definieron claves primarias, foráneas y check para tener un control de tipo de datos a ingresar en las tablas
  + Relaciones entre tablas
* Programación
  + Análisis y modelado de requerimientos como punto de partida inicial
    - Diagramas de secuencia
    - Diagramas de casos de uso
    - Diagrama de clases
      * Definir entidades
      * Definir relaciones
      * Definir métodos necesarios
  + Implementación
    - Lenguaje JAVA
      * J2SE
        + Inicialmente se inicia con Java Standard Edition o también conocido como J2Se para implementar la solución inicial
      * J2EE
        + Se avanza a una nueva etapa con Java Enterprise Edition o también conocido como J2EE
    - Interfaz gráfica
      * Se utiliza la biblioteca Swing
    - Conexión/Soporte persistencia
      * Se utiliza la API JDBC para conexión a base de datos como punto inicial
    - Escalabilidad
      * JBoss
        + Wildfly
* Testing
  + Quality control
  + Quality assurance
    - Se inició no solo realizando consultas por dudas en la documentación sino verificando no hubieran inconsistencias o errores.
  + Planificación de pruebas
    - Casos de prueba
    - Suite de prueba
      * Testlink
  + Registro de defecto y seguimiento de los mismos
    - Mantis

## 

## Marco Metodológico

* Base de datos
  + Se partió del MER de referencia, a partir de ahí se desarrolló el modelo físico considerando la estructura de tablas necesaria para soportar los requerimientos que teníamos asignados. A partir del modelo físico se crearon las tablas. Ya en la etapa de desarrollo fueron necesarios algunos cambios en el diseño de las tablas.
  + Para la gestión integral de la base de datos, se utilizó Oracle SQL Developer (versión 10g)
* Programación
  + Se utilizó documento UML para el modelado de entidades con sus atributos; métodos y relaciones
  + Se utilizó como herramienta de desarrollo Eclipse, es un software de tipo IDE (entorno de desarrollo)
  + Se utiliza Java Standard Edition para desarrollar la etapa inicial del proyecto el cual básicamente es una colección de APIs. Esto facilita el acceso a código reutilizable; métodos; entre otros.
  + Se utiliza Java Enterprise Edition (plataforma de programación) lo cual permite proveer mayor escalabilidad. Alguno de los beneficios a destacar son el uso de capas para mayor seguridad y escalabilidad; el tener la posibilidad de realizar desarrollo de lado de servidor y cliente de forma independiente.
  + Biblioteca Swing para poder desarrollar y diagramar la interfaz gráfica utilizada por el usuario
* Testing
  + Mantis, herramienta para reporte y gestión de incidencias.
  + Testlink, herramienta para reporte y gestión de casos de prueba

## Proceso realizado y Resultados

Se trató de que la distribución de trabajo fuera lo más razonable posible:

* Java - Noel, Martin y Camilo
* Base de Datos - Gonzalo
* Testing - Camilo y Gonzalo

Para las comunicaciones se utilizó:

* Google Drive - Almacenamiento de documentos (JAVA, BD, minutas, casos de prueba, documentos)
* Skype - Videoconferencias
* WhatsApp - Comunicación

Podemos definir el proceso realizado para este proyecto en 3 etapas:

Planeación:

Definición de flujos de casos de uso, diagramas y diseño de casos de prueba.

Definición:

Creación de clases JAVA y de objetos en base de datos.

Implementación:

Se implementan las funcionalidades propuestas en los requerimientos, para el apartado de programación se implementaron las clases java utilizando JPA para las entidades y EJB para la capa de negocios, en la interfaz visual se usó Swing. Como resultado logramos una aplicación que cuenta con una capa de negocios manejada por EJB y JPA, otra capa para la interfaz cliente. Para esta parte se trabajó asignándose una funcionalidad de implementar individualmente para luego subir a Google Drive y así reunir todo lo trabajado.

Con respecto a base de datos, se implementó según el modelado físico logrado a partir del MER provisto y cualquier actualización echa a la misma se subió a Google Drive acompañado del Workspace de Eclipse.

Para el apartado de Testing se definieron casos de prueba acorde a las funcionalidades a probar en la herramienta TestLink.

En esta etapa surge la dificultad de lograr la correcta implementación de JPA y una base de datos ya existente, logrando resolverla en poco tiempo, obteniendo así la correcta coexistencia entre entidades JPA y una BD existente.

## 

## Conclusiones

Los resultados más relevantes para el proyecto fueron la implementación de un sistema funcional que permite la creación de un alimento, la modificación y la asignación del consumo de un alimento a un ternero, agregar alimento al stock existente, así como también el alta y análisis de una enfermedad contraída por una ternera, desplegando en esta función el periodo de tiempo el cual estuvo enferma y el día de vida el cual contrajo la enfermedad entre los datos más relevantes, también cuenta con alta, baja y modificación de usuarios permitiendo filtrar ventanas por usuario, se implementó un histórico por enfermedad, peso y terneros. De esta forma logramos un nivel aceptable en el cumplimiento de los requerimientos, pudiendo así aconsejar a otro equipo que deba emprender un proyecto, a los cuales recomendamos

* Entender completamente los requerimientos antes de empezar a escribir código.
* Si lo estás buscando de una forma y no se logra, buscar alternativa, no hay una única forma de programar algo.
* Comunicación con los miembros del equipo constante.
* Cumplir tus objetivos asignados dentro del equipo.
* Re utilizar código siempre que sea conveniente
* Documentar cada cambio que se realice en la aplicación
* Respaldar antes de realizar cambios importantes ya sea en BD o Programación.

## Recomendaciones

Para continuar con este proyecto proponemos la implementación de las funcionalidades no contempladas en los requerimientos asignados, así como también una mejora en la interfaz para que sea aún más amigable al usuario, la implementación de auditoria en la base de datos para llevar el control de quien y cuando se realizan los cambios/registros.