**SAMI: Serious videogame of bovine cattle farms in Unity supported in System Dynamics**

Abstract

SAMI is a video game for learning about bovine cattle production systems. The game is developed with Unity Engine, has rules based on a model implemented with System Dynamics. On each game, the player must take decisions and do actions that allow him to properly run a farm. Those actions are stored in a website which, in order to ease the learning process, provides feedbacks by comparing the executed actions with those which are ideal. The article presents the main aspects of the proposal, that is, part of the model in which it is supported and a summary of the main aspects of implementation in Unity.

**Keywords:** SAMI, bovine cattle farms, Serious videogame, Unity, System Dynamics

**SAMI: Videojuego Serio de Ganadería bovina en Unity apoyado en Dinámica de Sistemas**

**Resumen**

SAMI es un videojuego para aprender sobre sistemas de producción de ganado bovino. El juego está desarrollado con Unity, tiene reglas basadas en un modelo implementado con Dinámica de Sistemas. En cada juego, el jugador debe tomar decisiones y realizar acciones que le permitan administrar una granja de forma adecuada. Esas acciones se almacenan en un sitio web que, para facilitar el aprendizaje, proporciona realimentación al comparar las acciones ejecutadas con las que son ideales. El artículo presenta los principales aspectos de la propuesta: parte del modelo en el que se basa y un resumen de los principales aspectos de la implementación en Unity.

**Palabras clave:** SAMI, Fincas de ganado bovino, videojuego serio, Unity, dinámica de sistemas

# Introduction

SAMI aims at teaching its players about the behavior and functioning of the bovine cattle production systems, by providing a close-to-reality interaction and offering feedback. This game not only provides players with enough resources and mechanics to play and stay engaged, but also helps them to develop skills in taking decisions about selecting the most appropriate breeds, feeding, milk production, treating diseases, buying and selling, among others.

# Contextual Framework

The following are the main topics comprised by SAMI:

**2.1. Cattle production systems**

In order to administer a cattle production system, several important parameters must be considered, regarding the amount of available resources and the status of the different related markets, among others (Phillips, 2003). It is also mandatory to understand the different complexities within the systems (Andrade et. al., 2001) in order to be able to propose and execute strategies for improving it.

**2.2. Simulation with System Dynamics (SD)**

System Dynamics is a methodology for learning, explaining and recreating phenomena of interest in terms of simulation models (Andrade et. al., 2001). It allows the observation of how the system behaves under different circumstances. It works under the assumption that the phenomena can be studied as dynamic systems, which can be explained and understood by making experiments of different situations (scenarios), and which shows the existing feedback that may exist among the variables of the model.

System Dynamics uses five languages of increasing complexity:

1. System verbalization: Explanation in natural language of the available knowledge about the phenomenon.
2. Causal loop diagram: Represents the system structure depending of the variables and the relations between them.
3. Stocks-flows diagram: Representation based on elements of System Dynamics, such as flows, level variables, parameters, auxiliary variables, exogenous variables and delays.
4. Mathematical equations: Linear or non-linear differential equations to evaluate the evolution in the time.
5. Model simulations: Visualizations of the data obtained after running the simulations.

**2.3. Serious games**

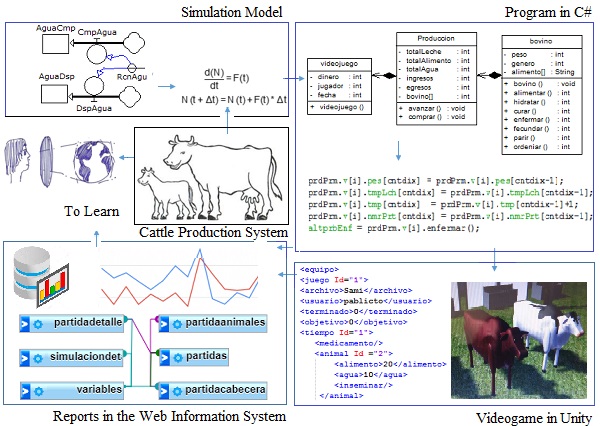
Serious games (SG) are an interactive entertainment-oriented instruments, which through controls allows to simulate experiences. In an electronic device, an SG is designed not only to entertain, but also to educate its players about some topic (Wein et. al., 2013). While usually similar to simulations, they aim to exploit the fun factor to enhance the learning process.

# Proposal

SAMI simulates several characteristics of the system, including feeding, growth, meat and milk production, breeding, health and death, and takes into account specific features of each breed. Players can easily interact with the system through an interface. The web page of the video game can be consulted in <http://intelec.bucaramanga.upb.edu.co/sami>.

The development of the game involves the six elements shown in Figure 1. The knowledge about the **Cattle Production System** is represented with System Dynamics resulting in a **Simulation Model** that support the game mechanics (these models represent a cattle farm that the player must administer by taking decisions to simulate experiences). The model is translated to a **C# program** that is then taken to the **Unity Engine.** The game along with the alerts, the player can view his results in a **web information system.** This feedback allows the player **to learn**, and at the same time provides a sense of progression that motivates continued effort. It is expected that there will be learning based on the feedback made by the user (the model is simulated in steps that correspond to months, and the player receives feedback for his decisions both during and after each month)

Figure 1. Components of the video game.



Source: Adapted (Gómez et. al., 2016).

## Model

The simulation model for cattle production is composed by five subsystems: demographic, biophysical, productive, financial and health (Gómez et. al., 2015). The subsystems are the basis for defining the equations that are implemented in C# for the game in the Unity framework. Decisions made by players during game-play are used for evaluation in each simulation step, for the game to provide feedback to the player.

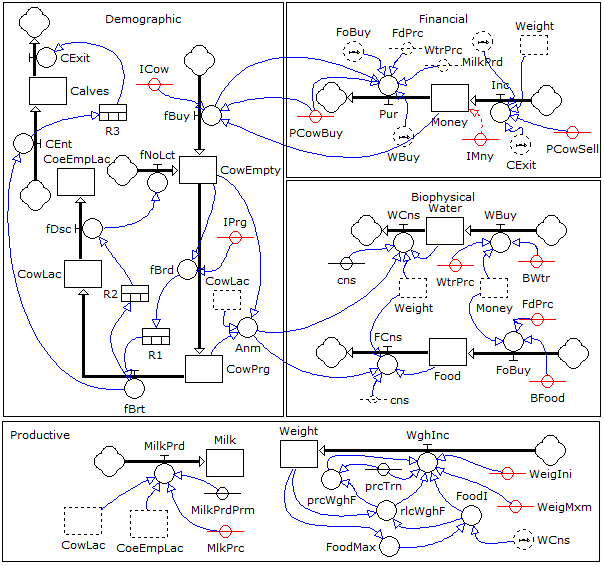
The stocks and flows diagram is shown in Figures 2. They show five feedback cycles and the main variables of the system. The analysis for the same is as follows:

1. The empty cows can be bred so they become pregnant cows, which give birth to calves and become lactating cows.
2. When the calves grow, they turn into empty cows, and the lactating cows are bred again.
3. At the same time, the calves and the empty cows and the milk can provide income if sold.
4. Any purchases decreases the amount of available money.
5. The purchase of food and water improves the weight of sales.

Some of the main variables within the system are the number of cows, the weight of each cow, the amount of water and food given to each cow and the available money. Stocks (rectangles) in Figure 2 represent them. These values can be changed by flows (circle-shaped valves), which represents events such as income, expenses, purchases, breeding and birth. The Stocks and flows diagram supports the equations for the simulation, observe the presence of auxiliaries, delays, clones and parameters (these are necessary to define the equations). The parameters allow you to set the constants with which the model will be used, such as:

* ICow: number of cows to buy
* PCowBuy: purchase price of cows
* IMoney: Money Initial
* PCowBuy: sell price of cows
* IPrg: number of cows to inseminate
* WtrPrc: purchase price of Water
* BWtr: Water to buy
* FdPrc: Price of Food
* BFood: Bood to buy
* MlkPrc: Sell Price of Milk
* WeigIni: Weigth inicitial of a Cow
* WeigMxn: Weigth maxim of a Cow

Figure 2. Stocks-flows Diagram of SAMI



For example, the following presents the equations of animal weight in each iteration:

FoodI = min (FoodMax, FoodI)

relWghF = FoodI / Weight \*100;

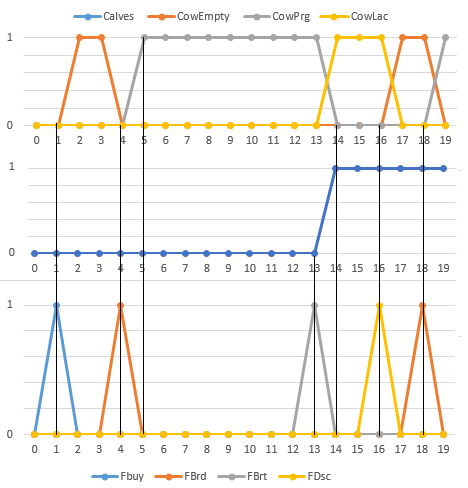
prcWghF = relWghF \* prcTrn

WghInc = prcWghF \* FoodI \* prcTrn

Weight = Weight + WghIc;

The figure 3 shows three graphs representing the behaviors of eight variables. The first two graphs corresponds to the levels (*Calves, CowEmpty, CowPrg, CowLac*) and the third, the flows (*FBuy, FBrd, FBrt, FDsc*). The graphs are there result of simulated parameters. Note that in the first month a *CowEmpty* was purchased and was inseminated in the 4th month, the gestation period ends in the 13th month; so, the *CowPrg* produces a Calf and goes to rest till the 16th month. During this period, the *CowLac* transissions to *CowEmpty* and again it is inseminated in the 18th month and the cycle repeats.

Figure 3. Model simulations with one Cow



## Implementation in Unity

After validating the model, it is implemented in Unity. The elements are created and they are programmed. Figure 4 shows SAMI in development mode from the platform, the indicated boxes cover the following elements:

1. Hierarchy: List of elements
2. Scene: Design view, you can see the sections where each animal moves and the initial presentation.
3. Initial Parameters: The current value of the cow in the market.
4. Inspector of assets: Allows access to edit the code.
5. Profiler: measures the performance of the RAM, memory, etc.
6. Visualizer of assets: The top pane shows the attributes and the bottom pane, the animation, in this case, a cow.

Figure 4. SAMI in Unity

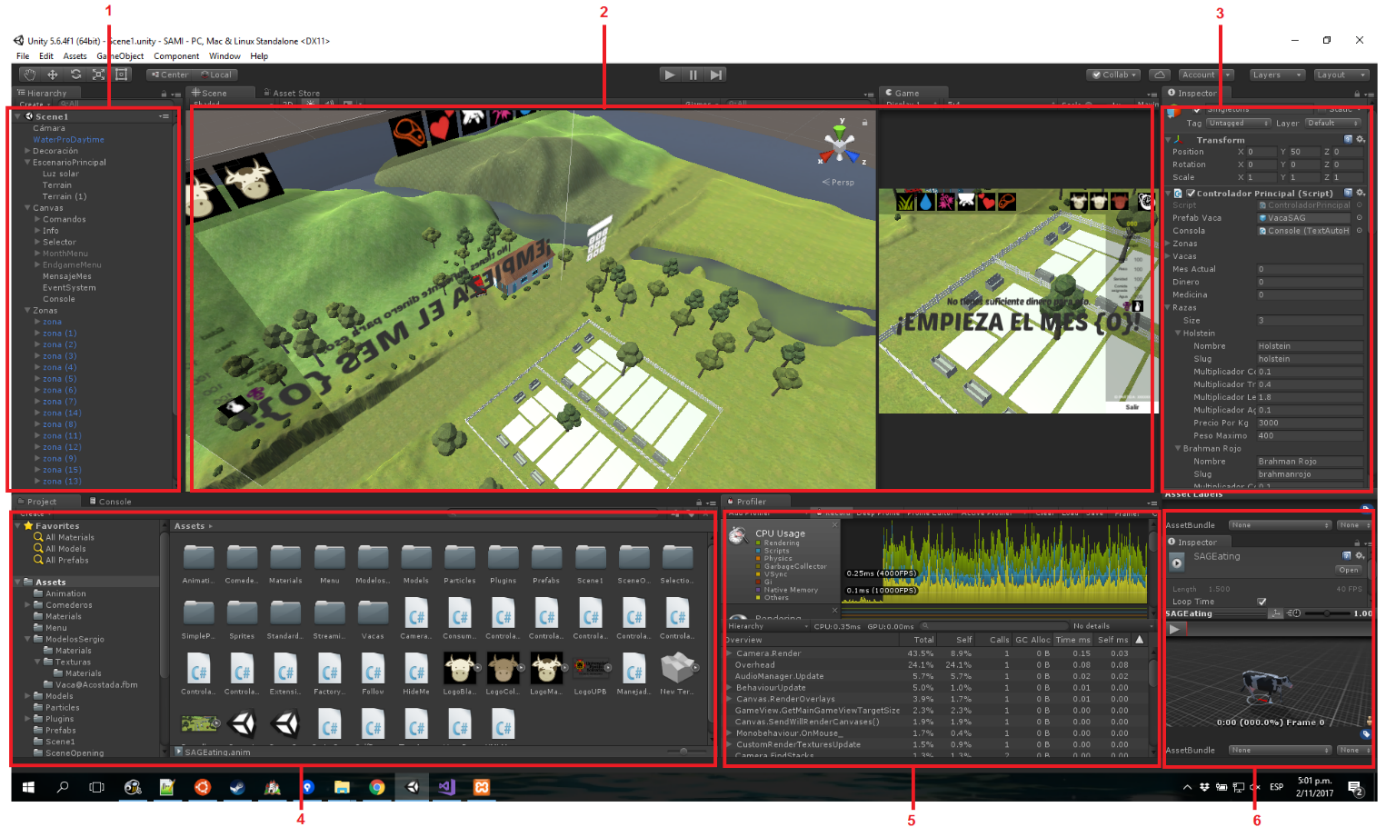


Figure 5 presents an example of programming a cow that corresponds to equations. Look at the lines 49, 50, 56, 58 and 59 of the code in C# presented (in the code of the program the names of the variables are more extensive).

Figure 6. Code in C#

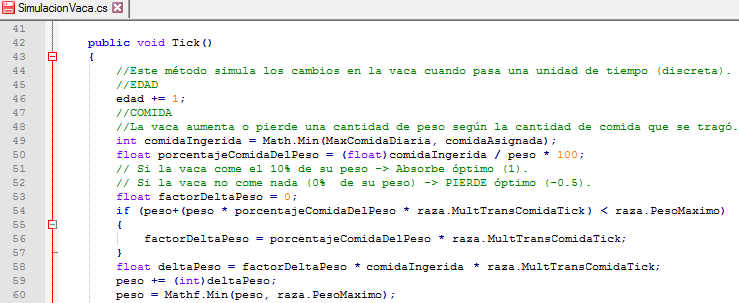


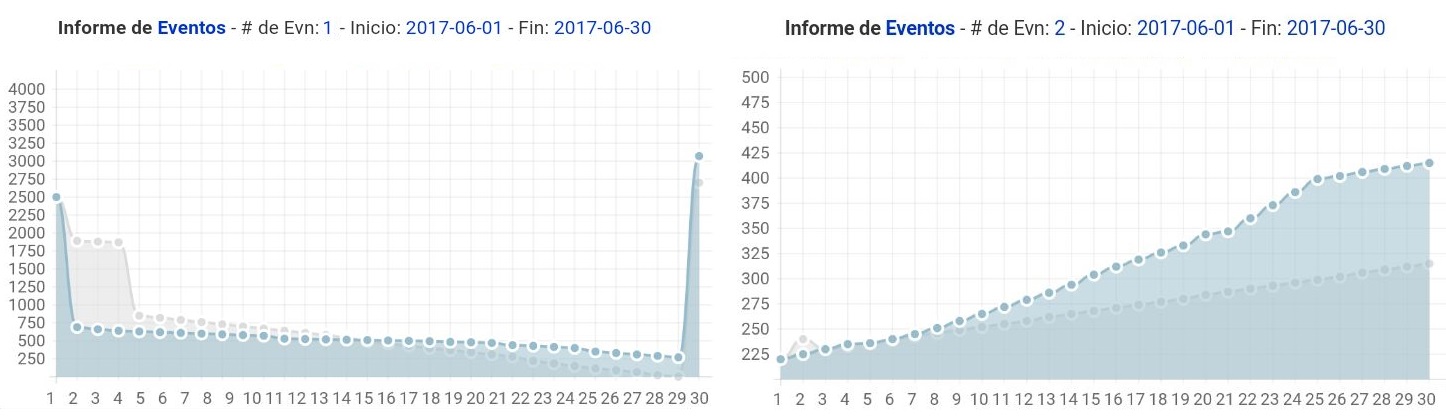
Figure 6 presents a farm with 17 animals. Observe the elements for decision making in the upper left part, which corresponds to providing food, water, medicine, milking, inseminate and to sell. The top right corner corresponds options to buy a Holstein, White Brahman and Red Brahman (species of cows) and to advance to the next month.

Figure 6. SAMI – Serious Game



Figure 7 shows one of the graphs that are presented from the web portal. In this case, the comparison of the money available in the game versus what should have been in case, correct decisions were made.

Figure 7. SAMI – Comparison of the money during a game



# Conclusions

Simulation models allow the execution of experiments with variations of their parameters, these “experiences” improve the knowledge (the initial budget, animal breeds and costs of buying and selling supplies) about managing a cattle farm.

In SAMI, the player understands the administration process of cattle productions systems during game-play. Players make an unconscious analysis of the dynamic system and the detection of feedback loops. The player performs experiments to analyze results generated with varied parameters. It is expected that they become aware of the aspects that improves administration of the farm. Tests have not yet been completed but it has been observed that students or apprentices have improved, taking better decisions with respect to cattle farming.

# References

# Andrade, H., Dyner, I., Espinosa, A., López, H., & Sotaquirá, R. (2001). *Pensamiento Sistémico, Diversidad en búsqueda de unidad*. Bucaramanga: UIS.

# Gomez Prada, U. E. & Gómez Sandoval, O. F. (2017). *Simulation of cattle farms with System Dynamics in a serious videogame. Case: SAMI*. Ponencia presentada en Big DSS Agro. Montevideo, Uruguay. 28 de septiembre.

# Gómez Prada, U., Andrade, H., & Vásquez, C. (2015). Lineamientos Metodológicos para construir Ambientes de Aprendizaje en Sistemas productivos agropecuarios soportados en Dinámica de Sistemas. *Infor. Tecnológica*, (4), 125-136.

# Phillips, C. (2003). *Principios de Producción Bovina*. Zaragoza: Acribia S.A.

# Wein, A.& Labiosa, W. (2013). *Serious games experiment toward agent-based simulation*. US Geological Survey Open File Report, 1152, 30p.