Revista UIS Ingenierías
ISSN: 1657-4583
ISSN: 2145-8456
revistaingenierias@uis.edu.co
Universidad Industrial de Santander
Colombia

Gómez-Prada, Urbano Eliécer; Gómez-Sandoval, Oscar Fernando
SAMI: Serious videogame of bovine cattle farms in Unity supported in System Dynamics
Revista UIS Ingenierías, vol. 18, núm. 4, 2019, Octubre-Diciembre, pp. 15-22
Universidad Industrial de Santander
Bucaramanga, Colombia

DOI: https://doi.org/10.18273/revuin.v18n4-2019001

Vol. 18, n.° 4, pp. 15-22, 2019
Revista UIS Ingenierías
Página de la revista: revistas.uis.edu.co/index.php/revistauisingenierias

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

SAMI: Videojuego serio de ganadería bovina en Unity apoyado en dinámica de sistemas

Urbano Gómez-Prada[1] Oscar Gómez-Sandoval[2]

[1]Grupo de investigación en Informática y Telecomunicaciones - Intelec, Facultad de Ingeniería de Sistemas e Informática, Universidad Pontificia Bolivariana, Colombia. Correo electrónico: [a] urbano.gomez@upb.edu.co, [b] oscar.gomezs@upb.edu.co

Received: 10 August 2018. Accepted: 14 February 2019. Final version: 16 July 2019.

Abstract

SAMI is a video game for learning about bovine cattle production systems. The game is developed with Unity Engine and 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 provides feedback by comparing the executed actions with those which are ideal in order to ease the learning process. 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.

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 proporciona realimentación al comparar las acciones ejecutadas con las que son ideales para facilitar el aprendizaje. 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.

1. 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 keep 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.

SAMI joins the assertion of [1] in terms of offering solutions to real problems through the design and development of simulations of situations allowing the strategies used by the player to generate conclusions about the production system.

ISSN Printed: 1657 - 4583, ISSN Online: 2145 - 8456, **CC BY-ND 4.0**

How to cite: U. Gómez-Prada, O. Gómez-Sandoval, “SAMI: Serious videogame of bovine cattle farms in Unity supported in System Dynamics,” *Rev. UIS Ing.*, vol. 18, no. 4, pp. 15-22, 2019. doi: 10.18273/revuin.v18n4-2019001

2. Contextual Framework

2.1. Cattle production systems

In order to manage 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 [2]. It is also mandatory to understand the different complexities within the system to be able to propose and execute improving strategies [3].

2.2. Simulation with System Dynamics (SD)

SD is a methodology for learning, explaining and recreating phenomena of interest in simulation models [3]. It allows to see how the system behaves under different situations. 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 reveal existing feedback loops among the variables of the model.

SAMI is supported in SD. This is because learning with the use of a simulation model can create the ability to adapt to complex changes in dynamic environments [4] by getting skills necessary to identify elements, relationships, feedback loops, and the selection of multiple alternatives [5].

2.3. Serious games (SG)

SG are interactive, entertainment-oriented instruments, which allow players to simulate and control experiences. In an electronic device, an SG is designed not only to entertain, but also to educate its players about some topic [6]. Although SG are usually used to run simulations, they aim at exploring the fun factor to enhance the learning process.

SAMI seeks the following objectives:

1. To provide a learning environment that allows players to experience, explore and discover multiple solutions without fear of making mistakes [7].
2. To engage the player with activities that generate learning from the selected decisions, to learn from mistakes, and to gain experience [8].
3. To develop strategic thinking and cognitive skills to assist decision making [9].
4. To generate skills and disseminate knowledge for training in various learning processes [10].

3. Proposal

SAMI simulates several characteristics of the system for six breeds: feeding, growth, meat and milk production, breeding, health and death. Players can easily interact with the system through an interface. In this paper are summarized the technical aspects of the model and its implementation in Unity. The web page can be consulted at http://intelec.bucaramanga.upb.edu.co/sami

The feedback cycle facilitates the learning of the bovine production system by improving the results in each game. The cycle occurs when a *bovine production system* is represented with a *simulation model* whose equations are implemented in a *video game.* The video game allows the players to check their successes and failures on a *web information system* (figure 1).

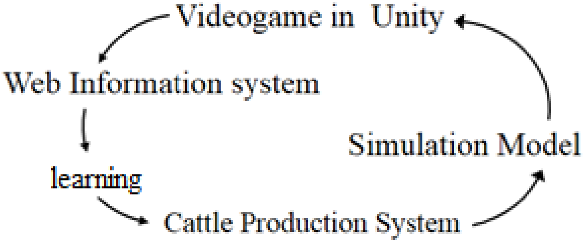


Figure 1. Learning Feedback Cycle. Source: Own elaboration.

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).

3.1. Simulation Model

The simulation model for cattle production is composed by five subsystems: demographic, biophysical, productive, financial and health [10]. The subsystems are the basis for defining the equations that are implemented in C# in the Unity framework. Decisions made by players during game-play are used for evaluation in each simulation step, to provide feedback.

The stocks and flows diagram are shown in Figure 2. They show five feedback cycles and the system main variables. The analysis is:

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, the empty cows and the milk can provide income if sold.

4. Any purchase 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.

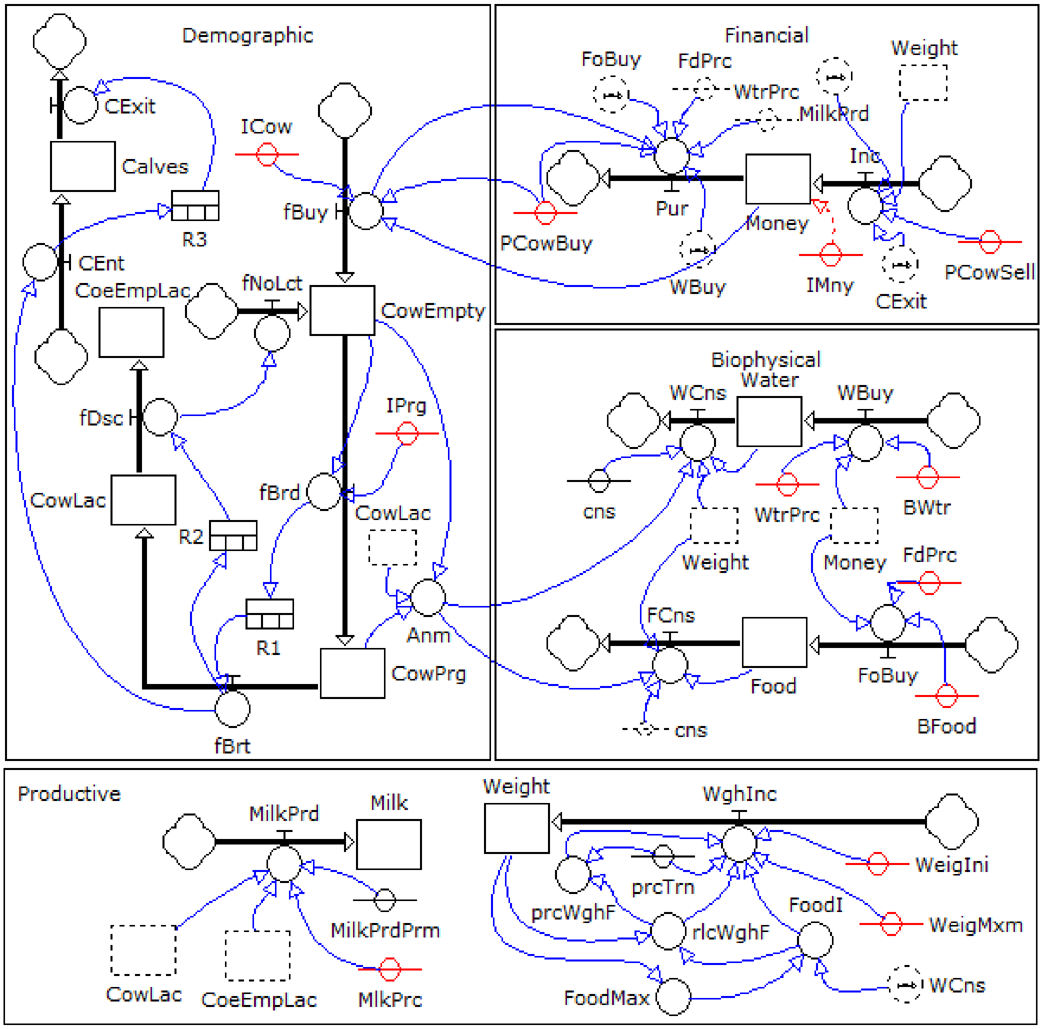


Figure 2. Stocks-flows Diagram of SAMI. Source: Own elaboration.

The stocks and flows diagram support the equations for the simulation, observe the presence of auxiliaries, delays, clones and parameters (these are necessary to define the equations). The parameters allow to set constants to 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: Food to buy
- MlkPrc: Sell Price of Milk
- WeigIni: Weight initial of a Cow
- WeigMxn: Weight maxim of a Cow.

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;

Figure 3 shows three graphs representing the behaviors of eight variables. The first two graphs correspond 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* makes a transition to *CowEmpty* and again it is inseminated in the 18th month and the cycle repeats.

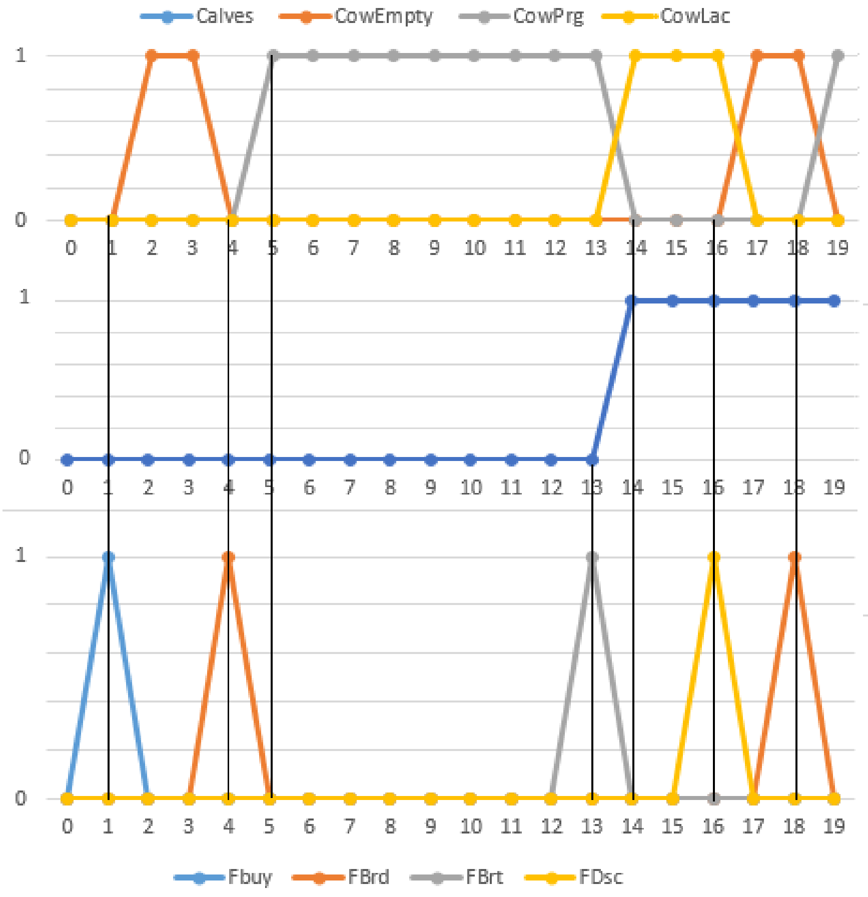


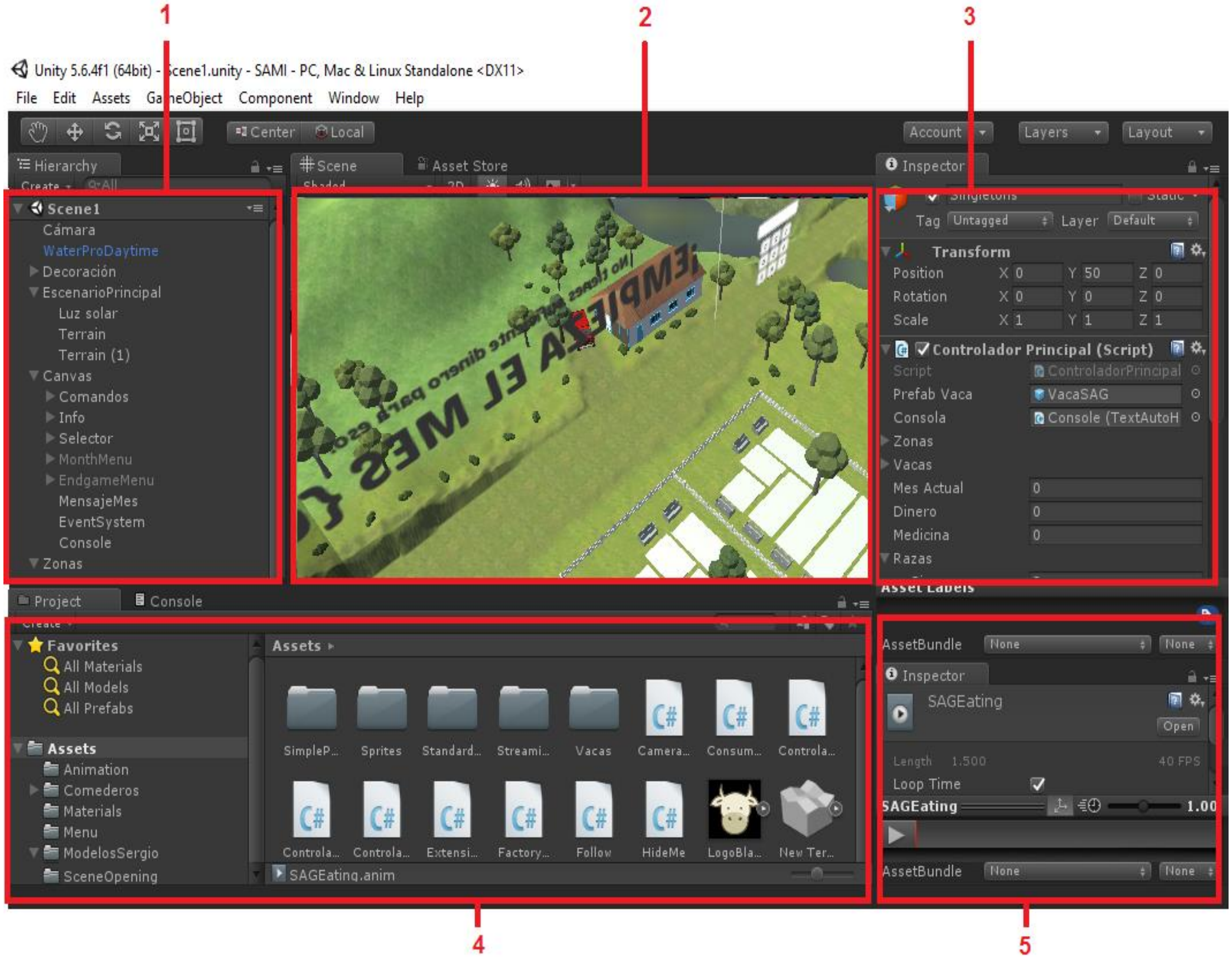
Figure 3. Model simulations with one Cow. Source: Own elaboration.

3.2. Implementation in Unity

The elements are created and programmed in Unity, a software that can be connected to three-dimensional objects (such as cows, stables, feeders, trees) which create elements that can be operated together and can be implemented with C# 6.0. Taking advantage of the cameras and lights offered by Unity, the interactions of elements generate scenarios which can be presented from different points of view, therefore, the experience of playing is more animated [13].

Figure 4 shows SAMI in development mode from the platform, the indicated boxes cover the following elements:

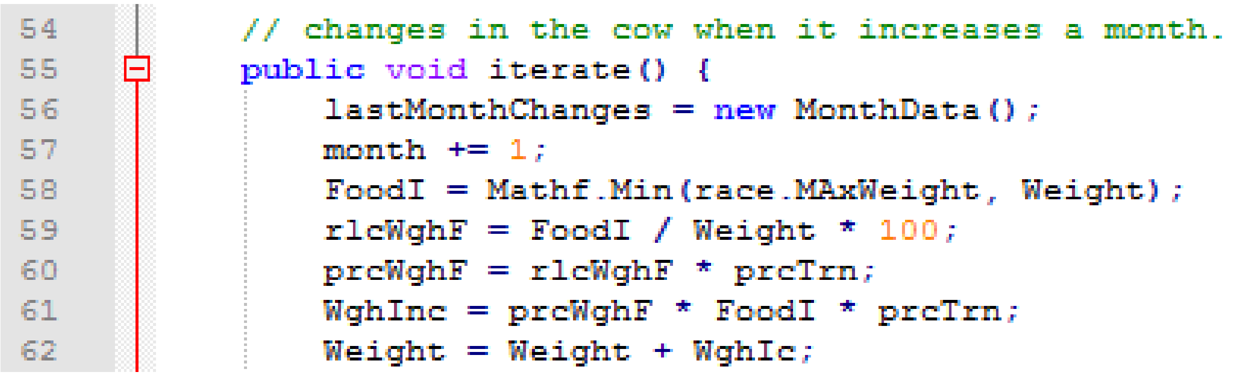
1. Hierarchy: List of elements
2. Scene: Design view, the player can see the sections where each animal moves and the initial presentation.
3. Initial Parameters: The current value of the cow in the market.
4. Inspector of assets: Access to edit the code.
5. Profiler: measures the performance of the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



Tag Untagged Layer Default
Transform
Position X 0 Y 50 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Controlador Principal (Script)
Script ControladorPrincipal
Prefab Vaca VacaSAG
Consola Console (TextAutoH
Zonas
Vacas
Mes Actual 0
Dinero 0
Medicina 0
Razas
AssetBundle None None
Length 1.500 40 FPS
Loop Time ☑
SAGEating 1.00
AssetBundle None None

Figure 4. SAMI in Unity. Source: Own elaboration.

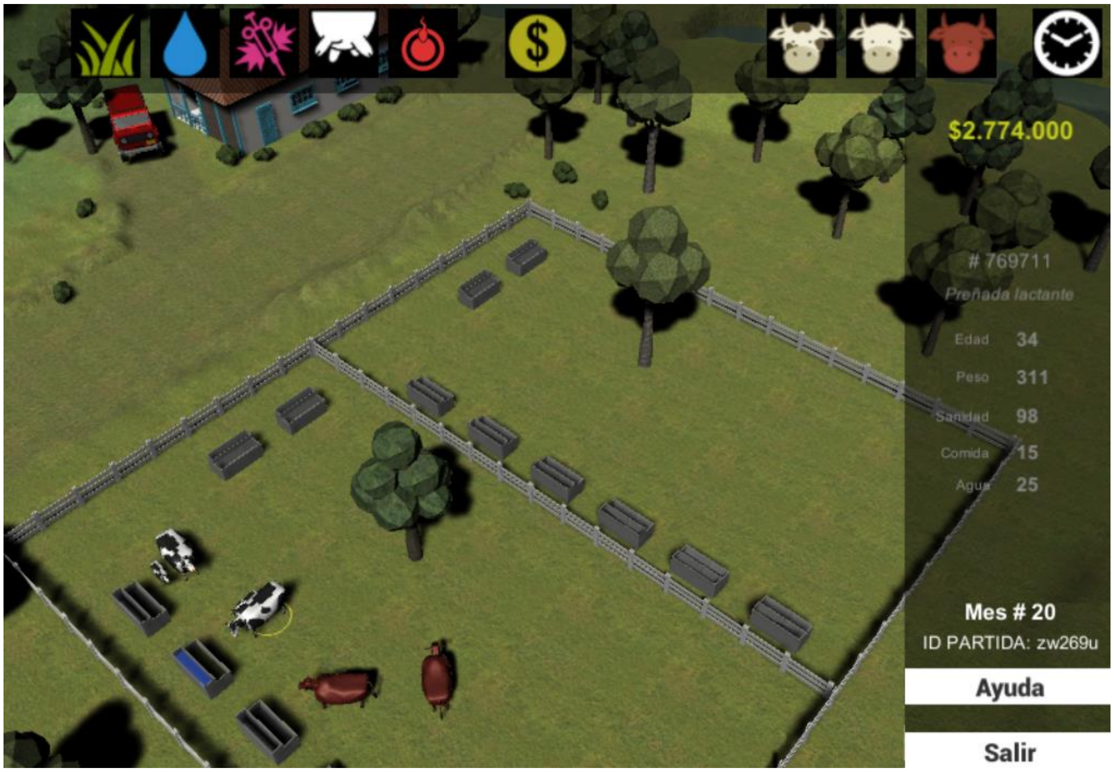
Figure 5 presents an example of programming a cow that corresponds to equations. Look at the lines 49, 50, 56, 58 and 59 of the code in C# presented (in the code of the program the names of the variables are more extensive).



```
54      // changes in the cow when it increases a month.
55      public void iterate() {
56          lastMonthChanges = new MonthData();
57          month += 1;
58          FoodI = Mathf.Min(race.MAxWeight, Weight);
59          rlcWghF = FoodI / Weight * 100;
60          prcWghF = rlcWghF * prcTrn;
61          WghInc = prcWghF * FoodI * prcTrn;
62          Weight = Weight + WghIc;
```

Figure 5. Code in C#. Source: Own elaboration.

Figure 6 presents a farm with four 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.



$2.774.000

769711

Preñada lactante

Edad 34

Peso 311

Sanidad 98

Comida 15

Agua 25

Mes # 20

ID PARTIDA: zw269u

Figure 6. SAMI – Serious Game. Source: Own elaboration.

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.

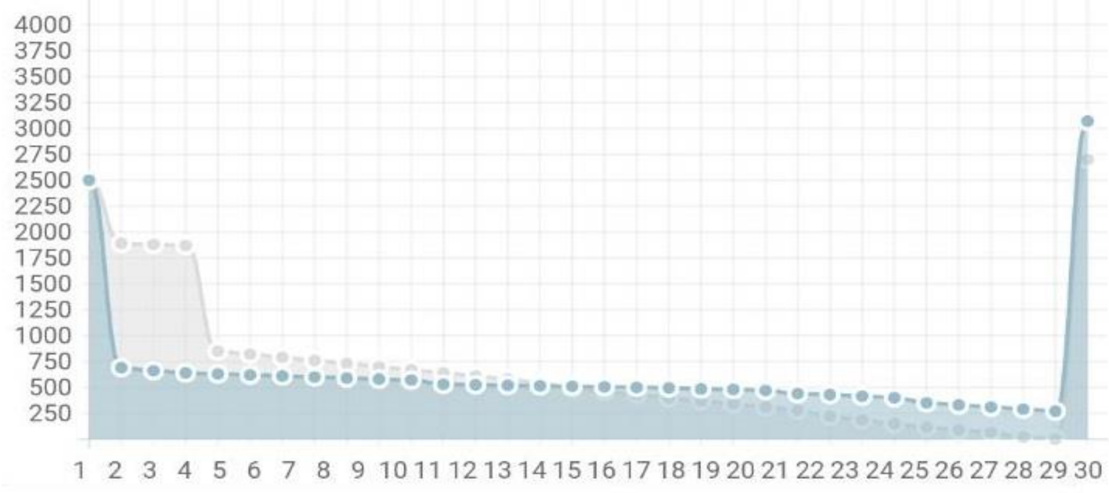


Figure 7. SAMI – Comparison of money. Source: Own elaboration

SAMI was used and validated by 11 sixth grade students of a rural school. These students live in cattle farms, so they are familiar with bovine livestock processes.

There were three guided sessions. In the first one, the simulation model was presented to unify concepts and to explain the videogame. During the remaining two sessions, apprentices used the SG to overcome challenges: the first one consisted in maintaining a single cow alive for 12 in-game months; the second one was to make a milk-producing farm with 10 cows, and to successively improve the production results of the virtual farm.

The sessions were guided by a tutor, who solves questions about how to use SAMI. During the first guided session, the tutor fixed some errors of the game related to calculations and sending data to the web portal. Those fixes were validated in the second session.

The following changes will be implemented in the next version of SAMI:

1. Extend the simulated model to include more significant variable costs. Although players have food and health expenses, new elements need to be added to express [12] expenses such as permanent labor and economic profitability calculations in milk and meat that display production costs.
2. Reduce of the digital gap of producers and replicating experiences in education as proposed [13] where favorable conditions were created for the change of practices in the use of ICT.

4. 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 the tests of SAMI has been observed that students during game-play: take better decisions with respect to cattle farming, understands the administration process of cattle productions systems, make an unconscious analysis of the dynamic system, detect of feedback loops and performs experiments to analyze results generated with varied parameters, so it is expected that they become aware of the aspects that improve administration of the farm.

References

[1] O. Boude figueredo and E. Sosa Neira, "Juego serio: modelo teórico para su diseño y producción," Universidad de la Sabana, 2016.

[2] C. J. C. Phillips and D. N. M. George, *Principios de producción bovina*. Acribia Editorial, 2003.

[3] H. Andrade, I. Dyner, A. Espinosa, H. López, and R. Sotaquirá, "Pensamiento sistémico: diversidad en búsqueda de unidad," Universidad Industrial de Santander, 2001.

[4] D. W. Ibarra Vega and J. M. Redondo, "Dinámica De Sistemas, Una Herramienta Para La Educación Ambiental En Ingeniería," *Rev. Luna Azul*, vol. 41, pp. 152–164, Jul. 2015.

[5] E. Pineda-Ballesteros, A. R. Lizcano-Dallos, and R. N. Lizcano-Reyes, "Redes sociales y representación de conocimiento como apoyo a los procesos de aprendizaje Social networks and knowledge representation as a support of learning processes," *Rev. UIS Ing.*, vol. 11, no. 2, pp. 237–252, 2012.

[6] A. Wein and W. Labiosa, "Serious games experiment toward agent-based simulation," Reston, VA, 2013.

[7] B. Gros Salvat, "Certezas e interrogantes acerca del uso de los videojuegos para el aprendizaje," *Rev. Int. Comun. Audiovisual, Public. y Estud. Cult.*, vol. 7, pp. 251–264, 2009.

[8] T. M. Connolly, E. A. Boyle, E. MacArthur, T. Hainey, and J. M. Boyle, "A systematic literature review of empirical evidence on computer games and serious games," *Comput. Educ.*, vol. 59, no. 2, pp. 661–686, Sep. 2012, doi: 10.1016/j.compedu.2012.03.004.

[9] M. Qian and K. R. Clark, "Game-based Learning and 21st century skills: A review of recent research," *Comput. Human Behav.*, vol. 63, pp. 50–58, Oct. 2016, doi:10.1016/j.chb.2016.05.023.

[10] C. López Raventós, "El videojuego como herramienta educativa. Posibilidades y problemáticas acerca de los serious games," *Apertura. Rev. innovación Educ.*, vol. 8, no. 1, 2016.

[11] U. E. Gómez Prada and O. Gómez S, "Simulation of cattle farms with System Dynamics in a serious videogame. Case: SAMI," in *Big DSS Agro 2017 (I Internacional Conference on Agro BigData and Decision Support systems in Agriculture)*, pp. 111–114.

[12] U. E. Gómez, H. H. Andrade, and C. A. Vásquez, "Lineamientos Metodológicos para construir Ambientes de Aprendizaje en Sistemas Productivos Agropecuarios soportados en Dinámica de Sistemas," *Información tecnológica,* vol. 26. scielocl, pp. 125–136, 2015.

[13] G. Méndez, E. Obviedo, G. Fallas, C. Vega, and A. Méndez, "Análisis de las herramientas Unity y Blender para el desarrollo de videojuegos con un enfoque educativo," in *V congreso de computación para el Desarrollo (COMPDES)*, 2012.

[14] H. Bolívar, J. Trocóniz, and A. Ruiz, "Diseño Y Evaluación De Una Estructura De Costos De La Ganadería Bovina En El Estado Barinas, Venezuela

Design And Evaluation Of A Cost Structure Of Bovine Raising In Barinas State, Venezuela Ciencias Básicas Y Tecnología,” *Saber*, vol. 28, no. 4, pp. 761–774, 2016.

[15] J. J. Moreno-Chaustre, H. H. Andrade-Sosa, J. R. García-González, U. Hernández-Pino, G. P. Maestre-Góngora, and G. López-Mólina, “Modelo de evaluación para valorar el cambio en las prácticas docentes con TIC. (Spanish),” *Rev. UIS Ing.*, vol. 13, no. 1, pp. 7–22, 2014, doi: 10.1016/j.sbspro.2012.05.139.