



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**  
**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**INFORME TÉCNICO**

**TEMA**

“DESARROLLO DE VIDEOJUEGOS CON TEXTURAS Y MODELOS 3D, MULTIUSUARIO (EN RED) Y EN UN ENTORNO CLIENTE/SERVIDOR (INTERNET) UTILIZANDO EL FRAMEWORK DE MICROSOFT XNA”

**APLICATIVO**

“VIDEOJUEGO PARA UN JUGADOR, MULTIJUGADOR (EN RED) E INTERNET”

**Autor:** Wilmer Carrera

**Director:** Ing. Mauricio Rea

**Ibarra – Ecuador**

**2012**

# “Desarrollo de videojuegos”

2012

Wilmer Manuel Carrera Pérez

Universidad Técnica del Norte

Facultad de Ingeniería en Ciencias Aplicadas

Carrera de Ingeniería en Sistemas Computacionales

Ibarra – Ecuador

will\_8818@hotmail.com

***Resumen.** El desarrollo de software de entretenimiento es prácticamente nulo en nuestro país, esto claramente se lo puede apreciar ya que ningún producto de este tipo tiene como sello de procedencia el nombre de Ecuador, una actividad que se ha convertido en una industria económicamente rentable a nivel mundial, lo cual nos convierte en un país netamente consumidor. Uno de los principales derivados son los videojuegos los cuales son acogidos por una gran población de consumidores.*

*El presente trabajo no pretende ser una guía de como desarrollar videojuegos con una herramienta específica, más bien trata de dar las pautas necesarias para conseguir este fin, ya que al ser novatos en este tema no se pueden establecer directrices formales u obligatorias que den como resultado un producto de calidad, los pasos a seguir se los tomo como experiencia de los productos desarrollados, los cuales fueron documentos de acuerdo al orden establecido por la metodología de desarrollo adoptada.*

## 1 Introducción

La industria de los videojuegos existe desde hace poco tiempo, y con el pasar de los años se ha consolidado como una de las industrias más activas y que más ganancias recibe.

En Ecuador el mercado de videojuegos está copado por software importado de este tipo, el desarrollo de videojuegos es casi nulo, convirtiéndonos así en un actor pasivo de esta actividad económica. Esta industria seguirá creciendo, y con el antecedente anterior, sería una buena opción participar activamente con productos nacionales de calidad.

El presente proyecto pretende documentar el desarrollo de un videojuego desde la concepción de la idea, hasta la culminación del mismo, cumpliendo todos los objetivos propuestos. El desarrollo de software de este tipo requiere conocimientos especializados de varias disciplinas. Debido a la complejidad que se afrontó en este proyecto el resultado final no es el de un videojuego profesional, realizado por compañías dedicadas a esta actividad.

### 1.1 Alcance

El desarrollo de videojuegos tiene un alcance a nivel mundial, obviamente que esto depende de que tan bueno sea el producto desarrollado, entrar en competencia con empresas dedicadas al desarrollo de

videojuegos sería una opción a largo plazo, para iniciar se debería irrumpir en la ciudad, luego según el éxito o fracaso tomar la decisión de continuar o abandonar esta actividad.

Se debe tomar en cuenta que para realizar esta actividad se necesita de personal bien capacitado y con conocimientos sobre algunos temas, ya que de optar por desarrollar videojuegos como una fuente económica sin conocer como afrontar un desarrollo de productos comerciales de este tipo los resultados serían negativos, por el contrario si desde el inicio se plantea objetivos bien definidos con nuevas e innovadoras ideas los resultados pueden ser sorprendentes.

En este punto es necesario mencionar que los videojuegos no son la única fuente de entretenimiento, con las herramientas investigadas se puede optar por desarrollos novedosos que actualmente no hay en el medio.

### 1.2 Posicionamiento

Actualmente son pocas las empresas en nuestro país que se dedican a esta labor, y los cuales en algunos casos son desconocidos a nivel nacional, al desarrollar productos de entretenimiento e insertarlos dentro de los consumidores locales se estaría ganando una buena participación localmente y según los

resultados, la posición de los productos se optaría por una zona mucho mas grande.

Con esta actividad no solo se conseguiría el desarrollo de videojuegos a gran escala si no también ser los pioneros a nivel nacional en productos innovadores que beneficien a la sociedad, los videojuegos no son malos si no mas bien son una oportunidad para mantener tanto al software como hardware actualizado, ya que cada vez más las aplicaciones de software consumen recursos que el hardware debe proveer para que la aplicación funcione correctamente, en el caso de los videojuegos o aplicaciones que consuman recursos gráficos, estos deben de presentarse al usuario de manera optima.

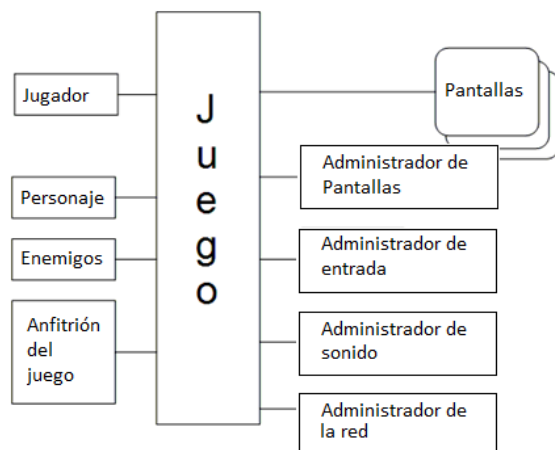
### 1.3 Definición del problema

El problema de	Escaso desarrollo de software de entretenimiento en nuestro país y no contar con centros de estudio especializado en esta actividad.
Que afecta a	Nadie en particular.
El impacto de ello es	El mercado nacional de consumidores de software de entretenimiento está copado por productos internacionales.
Una solución exitosa debería	Incursionar en esta actividad que es una de las actividades económicas más rentables a nivel mundial.

Aunque este problema no afecta a nadie, la solución beneficiaría a muchos, desde los creadores de software de entretenimiento hasta los consumidores, al ser motivados a utilizar productos locales.

### 1.4 Vista general del producto

Figura 1. Vista general del producto.



Fuente: El autor

Como el gráfico indica el juego se compone de varios módulos que se comunican entre si para dar como resultado un motor de videojuegos que según el tipo de videojuego este tendrá más funcionalidades dando jugabilidad y entretenimiento al usuario.

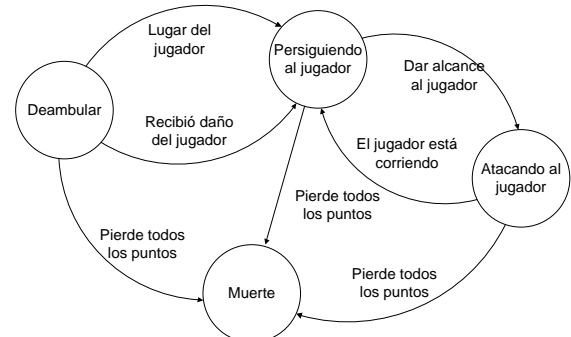
Los componentes en el grafico son solo los más básicos para conseguir un motor de videojuegos (Game Engine) profesional es necesario la utilización de componentes profesionales que hasta la actualidad están en proceso de ser de acceso libre.

En el desarrollo de los proyectos se utilizó una herramienta comercial, pero para fines no lucrativos existía una versión gratuita con limitaciones en su funcionamiento esta es el motor gráfico Sunburn, el cual al ser utilizado se logro mejorar la calidad en los gráficos no obstante, la versión profesional optimiza de mejor manera los efectos y gráficos que se puede aplicar a un escenario de videojuegos.

Los usuarios del videojuego que se los conoce como jugadores se comunican con la aplicación a través del teclado para que este tenga jugabilidad en el mundo 3D que se incorporo fue necesaria la implementación de personajes no controlados que actuaran como enemigos del jugador y que tienen relación con la historia del videojuego, en este punto para dar mayor realismo a la iteración jugador – aplicación fue necesaria la incorporación de efectos, es decir efectos como sistema de partículas y sonido por cada acción que puede presentarse en el funcionamiento del videojuego.

La inteligencia artificial aplicada para los objetos que no pueden ser controlados por el jugador es básica, controlada por un flujo basado en una maquina de estados, donde el personaje responde de acuerdo a eventos que puede ocurrir en el videojuego, como por ejemplo si el personaje es atacado este sigue a la fuente desde donde recibió el daño para atacarlo.

Figura 2. Diagrama de Inteligencia Artificial de los personajes no controlados



Fuente: El autor

Los demás componentes que intervienen en el motor de videojuego construido, en algunos casos son la implementación y modificación de componentes ya

existentes los cuales a medida que el proyecto lo requería fueron acoplados dando como resultado un motor casi completo en cuanto a funcionalidad pero muy pobre en cuanto a efectos especiales, si se comparase un videojuego profesional con los desarrollados, no tienen comparación, en la actualidad un software de entretenimiento de casas comerciales son obras de arte en cuanto a gráficos y demás opciones de juego.

En el campo del desarrollo de videojuegos intervienen algunas disciplinas que son necesarias conocer para aplicarlas correctamente y dar mejores resultados y más rápidos, requisito de esto y antes de empezar cualquier desarrollo es necesario conocer las estructuras matemáticas que la herramienta de desarrollo brinda, en el caso de los productos desarrollados el framework XNA proporciona una robusta API matemática para dar facilidad al programador a la hora de incorporar algún algoritmo, estas son vectores 2D obviamente para desplazarse en la pantalla en las coordenada X y Y, vectores 3D para interactuar con el escenario en X, Y y Z, matrices las cuales cabe mencionar XNA utiliza esta estructura para guardar todas las transformaciones que puede aplicarse a un objeto, como son escala, translación y rotación, esta información se guarda en matrices cuadráticas.

$$\begin{pmatrix} M_{11} & M_{12} & M_{13} & M_{14} \\ M_{21} & M_{22} & M_{23} & M_{24} \\ M_{31} & M_{32} & M_{33} & M_{34} \\ M_{41} & M_{42} & M_{43} & M_{44} \end{pmatrix}$$

Al aplicar una matriz identidad a un objeto que pertenece al mundo del videojuego este estará situado en la posición 0,0,0 sin rotación y con el mismo tamaño.

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

A lo largo del desarrollo de los sistemas propuestos, siempre se utilizó la matemática sobre estas estructuras para obtener una posición en pantalla, una coordenada, una proyección, o cualquier otro tipo de dato que serviría para aplicarlo en el videojuego.

En el sistema de física se puede claramente observar como se pueden aplicar métodos matemáticos para realizar cálculos que ayudan a la simulación de la física, de los revisado se pudo encontrar que los cálculos son de mucha complejidad llegando incluso al calculo de derivadas e integrales.

La versión en red para el producto desarrollado el cual soporta esta opción se escogió la topología cliente/servidor, esto debido a que se necesitaba lanzar la aplicación a internet la cual hace obvia por que se escogió este tipo de red.

Con las ventajas y desventajas que este tipo de red tenia las cuales son:

### Ventajas

Debido a que la simulación total del juego corre en una sola computadora, los juegos cliente/servidor son menos propensos a sufrir de problemas de consistencia.

Debido a que todas las decisiones se toman por parte de un solo servidor, es más difícil para los hackers engañar al modificar los archivos ejecutables del juego del cliente.

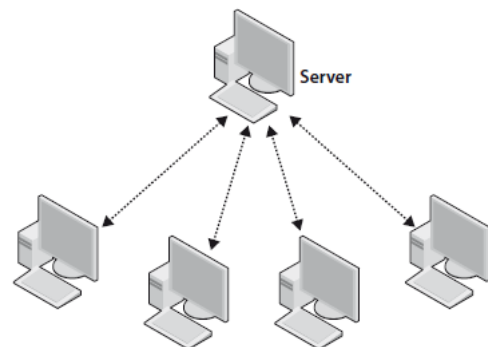
### Desventajas

El ancho de banda de red y la carga de trabajo que debe ser procesada por la CPU provocaría una caída en el servidor, que fácilmente puede convertirse en un cuello de botella en el procesamiento.

La red de ida y vuelta desde y hacia el servidor puede hacer que los jugadores en los equipos cliente a experimentar el retraso en el juego al responder sus peticiones. (El servidor no sufre este retraso, por lo que es una "ventaja del anfitrión", comportamiento que puede ser observado en muchos juegos cliente/servidor.)

Debido a que el estado de simulación sólo existe en un solo equipo, el soporte de migración del servidor es más difícil que estar usando una red.

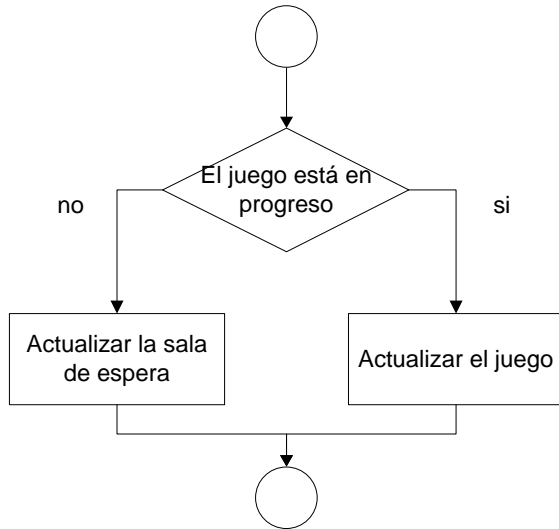
Figura 3. Típica red Cliente/Servidor.



Fuente: El autor

Para este sistema se eligió la funcionalidad del videojuego Quake, para conseguir esto fue necesaria la modificación del mecanismo de unión de jugadores a la sala de juego es decir cuando el servidor inicia la sesión crea un enlace para que los demás jugadores ingresen a la sesión abierta donde el anfitrión es quien inicia la red, luego de que toda la sala esta logueado en la sesión deben de marcarse como listos y posterior a esto el anfitrión inicia el videojuego, con lo anteriormente manifestado el mecanismo que por defecto XNA provee no soporta el ingreso de nuevos jugadores luego de que el videojuego es iniciado, en este punto la sesión cierra la conexión para jugadores nuevos.

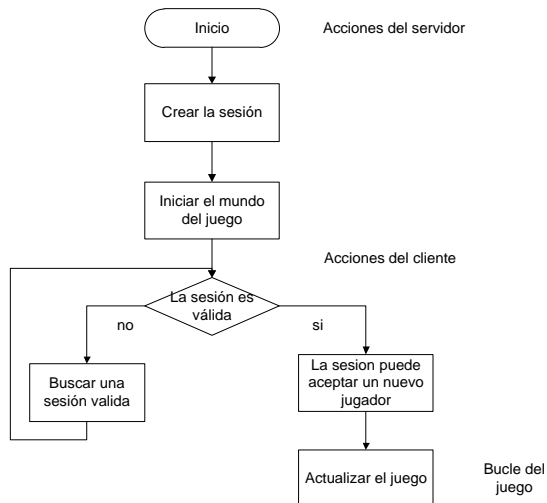
**Figura 4.** Mecanismo por defecto que XNA provee para la unión de jugadores.



**Fuente:** El autor.

Como puede notarse en el diagrama de flujo si la sesión no esta en estado jugando (playing) esta actualizado la sala de espera donde los jugadores se marcan como listos (ready) caso contrario esta actualizando el juego cuando la sesión es cerrada pero anterior a esto se unieron jugadores a la sesión.

**Figura 5.** Mecanismo modificado para unir jugadores cuando el videojuego esta en ejecución.



**Fuente:** El autor.

El gráfico es de lo más sencillo, el anfitrión de la sesión es decir el servidor, inicia una sesión directamente crea el mundo del videojuego en este caso habría un participante en la sesión de red, esperando que alguien más ingrese para dar comienzo al videojuego, en este caso no hay una sala de espera donde los jugadores esperan a marcarse como listos para luego ser ingresados, el mecanismo desarrollado hace que un jugador que haya encontrado una sesión válida y este escoja esa sesión, directamente es

ingresado al videojuego, todas las propiedades del mundo hasta el momento que ingresa son enviadas al nuevo jugador para que este los procese y cree un mundo sincronizado con respecto a los demás jugadores. La comunicación a través de la red entre los integrantes de la sesión de red consta en la versión completa del presente trabajo.

En redes siempre existen inconvenientes que no se puede controlar como: ancho de banda, latencia y pérdida de paquetes, estos son causados por los medios físicos que compone la red, lo cual hace de la programación de videojuegos en red una tarea ardua.

Según la documentación de XNA, el framework soporta hasta 16 clientes conectados, pero solo se probó con 4 clientes esto debido a la disponibilidad de equipos

La versión en internet es la misma que la versión en red, debido al costo de un servidor dedicado para videojuegos y las configuración sobre dicho servidor para que sea publico, se creo una VPN para su uso lo cual cabe indicar solo se realizo entre 2 computadores debido a que no eran conexiones estables, la velocidad era baja y a cada momento se perdía la conexión.

Como resultado de la incorporación de todo lo antes expuesto se consiguió crear 2 videojuegos con el nombre de:

**Figura 6.** Splash screen para el videojuego de un solo jugador.



**Fuente:** El autor.

**Figura 7.** Splash screen para el videojuego en red.



**Fuente:** El autor.

Para cada uno de estos proyectos realizados se creo un instalador de uso libre, sin restricciones, con las respectivas instrucciones de como instalar la aplicación no se necesita de un manual de usuario ya que para estos tipos de software el uso es por demás intuitiva y agradable.

Los instaladores, fuentes y recursos del proyecto pueden ser obtenidos en el DVD que acompaña este proyecto, queda por demás manifestar que desarrollar esto fue bastante gratificante y muy provechoso, por todo el conocimiento adquirido.

### 3 Conclusiones

Se consiguió completar las versiones de los videojuegos propuestos en su totalidad y superando las expectativas iniciales del proyecto ya que en un principio para la versión para un solo jugador no se planteó la posibilidad de utilizar vehículos con comportamiento físico, efectos de sistemas de partículas, luces ni sombras, y para la versión multijugador nunca se pensó en la posibilidad de incorporar un chat entre los jugadores y por el desconocimiento inicial de framework no se conocía que se podía transmitir streaming de voz a través de la red, simulación de trafico de red, únicamente instanciado un componente, estas y otras características que se pueden ver y probar en los videojuegos desarrollados que dan como resultado un producto con mayores prestaciones de calidad y presentación.

Los estándares de calidad para este tipo de software no están definidos por las compañías o personas que se dediquen a la creación de videojuegos, si no más bien quien decide si un producto de estos es de calidad, es el usuario que utilice el producto, debido a esto, uno de los objetivos planteados no puede ser completado o definido por lo antes expuesto, además al ser novatos en este campo no se puede definir reglas que se deben seguir para entregar un producto final de calidad, si bien hasta ahora con las compañías existentes no hay normas publicadas para el desarrollo de videojuegos, con este antecedente no se tiene la autoridad suficiente como para dar reglas o como fue expuesto estándares de calidad sobre un producto de entretenimiento, por la razón antes citada sobre el ser novatos, fue también un motivo por lo que se estableció como objetivo el entregar como resultado estándares de calidad que como se manifestó no pudo ser completado con las aclaraciones pertinentes.

Dada la complejidad del desarrollo de un videojuego, en algunos casos fue necesaria la adaptación de elementos ya existentes y de acuerdo a lo que el proyecto requería estos fueron modificados.

Con el esfuerzo necesario y aprendizaje metodológico del framework XNA, se puede lograr desarrollar un videojuego completo y avanzado.

El motor de videojuegos es extensible y personalizable para videojuegos 3D, pero solo del

tipo: disparos en primera persona y en tercera persona.

### 4 Recomendaciones

XNA no solo se lo puede utilizar en el desarrollo de videojuegos, aunque sea un framework para este fin, a partir de este se puede derivar muchos otros productos como: sistemas multimedia, aplicaciones para dispositivos móviles, etc. Un mercado que en el país no es muy explotado.

Hasta la elaboración del presente documento, está en auge los dispositivos con soporte para Windows Phone 7, implementaciones de aplicaciones XNA para silverlight, dispositivos de hardware como Tablets Pc, aplicaciones Touch Screen, entre otras (inicios del año 2012), con los conocimientos necesarios sobre este framework no solo se desarrollaría para PC sino también para las plataformas antes mencionadas con lo cual se incursionaría sobre un mercado nuevo con grandes beneficios económicos.

Desarrollar aplicaciones con XNA requiere cierto grado de habilidad al programar y un conocimiento medianamente avanzado del lenguaje de programación, en este caso C#, habilidades que se logró adquirir durante el desarrollo del sistema, resaltando que al inicio del proyecto el conocimiento era básico, es decir, se conocía el lenguaje pero no todo su potencial además el desconocimiento del desarrollo de sistemas de entretenimiento, con lo que resulta recomendable probar nuevas tecnologías de desarrollo para aumentar y mejorar los conocimientos.

### Referencias

- [1] Curtis Bennett, A Simple Introduction to Game Programming With C# and XNA 3.1, First Edition: 2009, Estados Unidos de America.
- [2] Jerry Lee Ford, Jr. XNA 3.1 Game Development for Teens, Course Technology PTR A part of engage Learning, Estados Unidos de America.
- [3] David. M Bourg, Physics for Game Developers, Primera edición, O'Reilly, Estados Unidos de America, Enero 2002.
- [4] Aaron Reed, Learning XNA 3.0, Primera edición, O'Reilly, Estados Unidos de América, Noviembre 2008.
- [5] Aaron Reed, XNA 3.0 Unleashed, Primera edición, O'Reilly, Estados Unidos de América, Enero 2009.
- [6] Rob Miles, Introduction to Programming Through Game Development Using Microsoft XNA Game, Primera edición 2010, Microsoft Press, Estados Unidos de América.
- [7] Rob Miles, C# Development, Edición 1.1, University of Hull, Estados Unidos de América, Octubre 2009.

- [8] Riemer Grootjans, XNA 3.0 Game Programming Recipes, Primera edición 2009, Apress, Estados Unidos de América.
- [9] Alexandre Santos Lobão, Bruno Evangelista, José Antonio Leal de Farias, and Riemer Grootjans, Beginning XNA 3.0 Game Programming From Novice to Professional, Primera edición 2009, Apress, Estados Unidos de América.



**TECHNICAL OF THE NORTH UNIVERSITY  
FACULTY OF APPLIED SCIENCE ENGINEERING  
ENGINEERING IN COMPUTER SYSTEM**

**TECHNICAL REPORT**

**TOPIC**

“GAME DEVELOPMENT WITH TEXTURES AND 3D MODELS, MULTIPLAYER (NETWORK) AND CLIENT/SERVER ENVIRONMENT (INTERNET) WITH THE MICROSOFT FRAMEWORK XNA”

**APPLY**

“VIDEO GAME FOR A SINGLE PLAYER, MULTIPLAYER (NETWORK) AND INTERNET”

**Author:** Wilmer Carrera

**Director:** Ing. Mauricio Rea

**Ibarra – Ecuador**

**2012**



# “Video game development”

2012

Wilmer Manuel Carrera Pérez

Technical of the North University  
Faculty of Applied Science Engineering  
Engineering in Computer Systems

Ibarra – Ecuador

will\_8818@hotmail.com

*Summary Software entertainment development is virtually null in our country, that it can be seen clearly because no such product has a seal of origin the name of Ecuador, an activity that has become an economically viable industry worldwide , which makes us a country purely consumer. One of the main derivatives are the video games which are hosted by a large population of consumers.*

*The present work not intended as a guide to develop video games with a specific tool, rather tries to give the necessary guidelines to achieve this, since being new to this topic cannot be established formal guidelines or mandatory that result a quality product, the steps to follow was taking as an experience of the developed products, which were documents in the order established by the development methodology adopted.*

## 1 Introduction

The video game industry has existed for some time, and with the passing of the years has become one of the world most active and get more profit.

In Ecuador, the video game market is full by imported software this type, game development is almost zero, thus becoming a passive actor in this business. This industry will continue to grow, and the previous antecedent, would be a good option to participate actively with national quality products.

This project aims to document the progress of a video game from idea conception to the completion of it, meeting all objectives. The development of this type software requires expertise from various disciplines. Due to the complexity of this project was tackled in the end result not is a professional video game, made by companies engaged in this activity.

### 1.1 Scope

The video games development has a reach world level, obviously this depends on how good the product developed and competing with companies involved in video game development would be a long-term option to start should into the local city, then with the success or failure to take the decision to continue or abandon this activity.

It should be noted that for this activity will need well-trained and knowledgeable about some topics, and that choose to develop video games as an economical source without knowing how to deal with development of commercial products this type, the results would be negative, on the contrary if from the outset clear objectives arises with new and innovative ideas the results can be surprising.

At this point it should be mentioned that video games are not the only source of entertainment, with the tools investigated other option as application for mobile devices.

### 1.2 Positioning

Currently there are few companies in our country engaged in this work, and which in some cases are unknown nationally, to develop entertainment products and insert them into the local consumers would be earning a good turnout locally and according to the results the position of the products opt for a much larger area.

This activity not only get the video game development on a large scale, also to pioneer nationally in innovative products that benefit society, video games are not evil but rather as an opportunity to maintain software and hardware updated, because every time more software applications consume resources that the hardware must provide for the application to work correctly, in the case of video games or graphics applications

that consume graphics resources, they must be presented to the user in an optimal manner.

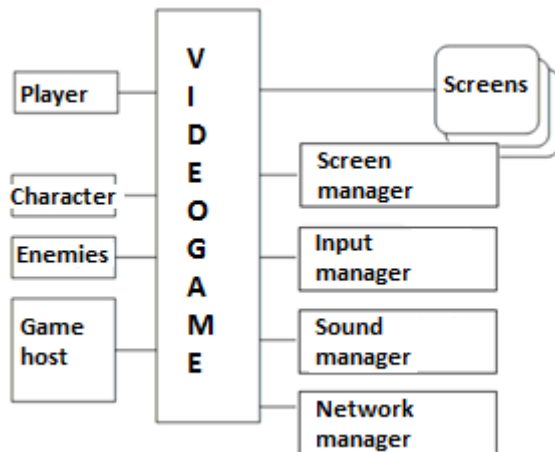
### 1.3 Defining the problem

The problem of	Weak development of entertainment software in our country and not having study centers specializing in this activity.
It affects	No one in particular.
The impact of this is	The market national consumer for entertainment software is full by international products.
A successful solution should	Venture into this activity is one of the activities most profitable economic worldwide.

Although this problem does not affect anyone, the solution would benefit everyone from the creators of entertainment software until consumers, to be encouraged to use local products.

### 1.4 Product Overview

Figure 1. Product Overview



Source: The author

As the graph indicates the game is composed of several modules that communicate with each other to result in a video game engine based on the type of game it will have more functionality giving the user playability and entertainment.

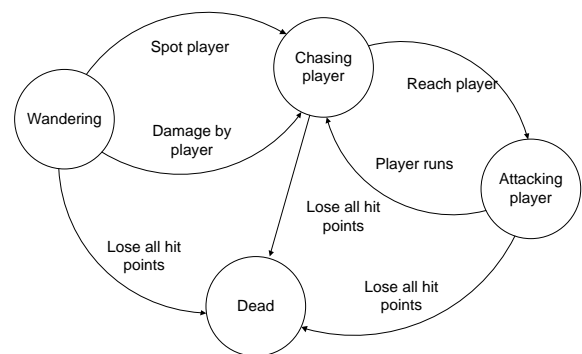
The components in the graph are just the most basic, for a professional Game Engine is necessary to use components to professionals who are currently in the process of open access.

In the development of the projects use a commercial tool, but for nonprofit was a free version with limitations in performance this is the Sunburn graphics engine, which when used to improve the quality of the graphics however, professional version optimized for best effects and graphics that can be applied to a video game scenario.

Users of a video game are known as players communicate with the application through the keyboard so that it has gameplay in a 3D world that was incorporated necessary to implement not controlled characters act as enemies of the player and that relate to video game history, at this point to give more realism to the iteration player - application was necessary to incorporate effects, like particle systems and sound for every action that may arise in the operation of the game.

Artificial intelligence applied to objects that cannot be controlled by the player is basic, controlled by a flow based on a state machine, where the character responds according to events that could occur in the video game, such as whether the character is attacked it follows the source from received damage to attack.

Figure 2. Diagram of AI not controlled characters



Source: The author

Other components involved in the game engine built, in some cases are the implementation and modification of existing components which as the project required it were coupled motor resulting in almost complete in functionality but very poor in as for special effects, compare it if a professional video game with the developed, no comparison!, today's entertainment software business houses are works of art in terms of graphics and other game options.

In the field of video game development meddle some disciplines that are necessary to know to apply them properly and give better results and faster, and this requirement before any development is necessary to know the mathematical structures

that provides development tool, in the case of products developed on XNA framework provides a robust API to ease the mathematical programmer at the time to incorporate some algorithm, these are vectors 2D obviously to scroll on the screen in the X and Y coordinates, vectors 3D to interact with the scenario in X, Y and Z matrices which include XNA uses this structure to save all the changes that can be applied to an object, such as scale, translation and rotation, this information is stored in quadratic matrices.

$$\begin{pmatrix} M_{11} & M_{12} & M_{13} & M_{14} \\ M_{21} & M_{22} & M_{23} & M_{24} \\ M_{31} & M_{32} & M_{33} & M_{34} \\ M_{41} & M_{42} & M_{43} & M_{44} \end{pmatrix}$$

Applying an identity matrix to an object that belongs to the world of the video game this will be located at the 0,0,0 position without rotation and with the same size.

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Throughout the development of the proposed systems, always use the math on these structures to obtain a position on the screen, a coordinate, a projection, or any other data type that would serve to implement in the video game.

In the physical system could clearly see how you can apply mathematical methods to perform calculations that help the physics simulation, the revised could be found that the calculations are very complex even to the calculation of derivatives and integrals.

The network version for the developed product which supports this option was chosen topology client/server, this because was needed the send the application to internet which makes it obvious that you chose this type of network.

With the advantages and disadvantages of this type network which are:

**Advantage**

Because the total simulation of the game is at a single computer, game client / server are less likely to suffer from problems of consistency.

Because all decisions are made by a single server, it is more difficult for hackers to cheat by modifying the executable files of the game client.

**Disadvantages**

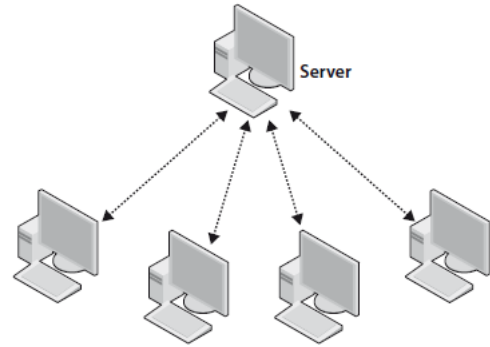
The bandwidth and network workload to be processed by the CPU cause a drop in the server, which can easily become a bottleneck in processing.

The network round trip to and from the server can make the players on the client computers to experience delays in the game to answer your requests. (The server does not suffer this delay, so it

is a "host advantage" behavior can be observed in many games client/server.)

Because the simulation state exists only in a single computer, server migration support is more difficult to be using a network.

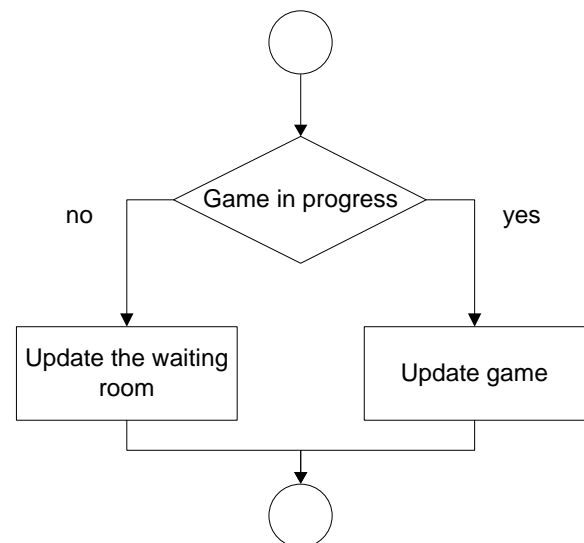
**Figure 3.** Client/Server network



**Source:** The author

For this system was chosen the functionality of the video game Quake, to achieve this was necessary to modify the mechanism of binding players to the game room that is when the server starts the session creates a link to the other players entering the open session where the host is who started the network, after the entire room is logged in the session should be marked as ready and after the host start the game, with the above stated, the default mechanism that does not support XNA provides not income of new players after the game is started, at this point the session closes the connection for new players.

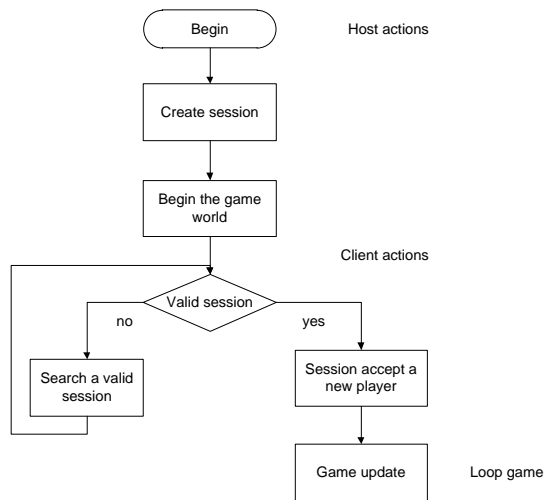
**Figure 4.** Default XNA mechanism that provides for the players' union.



**Source:** The author

As can be seen in the flow chart if the session state is not playing, this updating the waiting room where players are marked as ready, on the contrary is updating the game when the session is closed but before this players joined to the session.

**Figure 5.** Modified mechanism for joining players when the game is running.



**Source:** The author

The graph is very simple, the host of the session, the server begin the session directly creates the video game world in this case would be an active participant in the network, waiting that someone enter to start the game, in this case not waiting room where players expect to be marked as ready to be admitted then, the developed mechanism makes a player who has found a valid session and choose the session, is entered directly to the game, all properties the world until the moment you enter are sent to new player for this to process and create a world synchronized with respect to the other players. Communication over the network between the members of the network session consists in the full version of this work.

In networks there are always problems that cannot be controlled as bandwidth, latency and packet loss, these are caused by physical means making up the network, which makes the network game programming a challenging task

According to the XNA documentation, the framework supports up to 16 connected clients, but only tested with 4 clients this because of the availability of equipment.

The online version is the same as the network version, due to the cost of a dedicated server for game and the settings on the server to be public, was created for use a VPN which can be indicated only took place between 2 computers because they were not stable connections, the speed was low and each time the connection was lost.

As result of incorporating all of the above is able to create 2 games with the name:

**Figure 6.** Splash screen for a single player video game.



**Source:** The author

**Figure 7.** Splash screen for network video game.



**Source:** The author

For each of these projects was created an installer for free use, without restrictions, with the respective instructions how to install the application does not need a user manual because these types of software the use is intuitive and enjoyable.

Installers, sources and resources of the project can be found on the DVD that accompanies this project, only enough manifest that develops this was rewarding and profitable, for all the knowledge acquired.

### 3 Conclusions

Achieve complete the versions of the game proposed in its entirety and exceeding initial expectations of the project as a principle for the single player version was not raised the possibility of using vehicles with physical behavior, effects of particle systems, lighting or shadows, and the multiplayer version never thought about the possibility of incorporating a chat between the players and the initial ignorance was not known framework that could transmit voice streaming over the network, simulation of network traffic,

instantiated only one component, these and other features that you can see and taste in video games developed resulting in a product with higher quality services and presentation

The quality standards for this type of software are not defined by companies or persons involved in the creation of video games, but rather decide whether a product is quality of these is the user who uses the product, for this reason, one of the objectives cannot be completed or defined by the above, in addition to being new to this field cannot define rules to follow to deliver a quality product, but so far with companies no existing published standards for game development, with this antecedent does not have enough authority to make rules or standards of quality was exposed on an entertainment product for the reason cited above about being rookies, was also a reason that the target was established as a result deliver quality standards as stated could not be completed with the necessary clarifications.

Given the complexity of video game development, in some cases it was necessary to adapt existing elements and according to what the project required these were modified.

With the effort needed and methodological learning XNA framework, can achieve full develop a video game and advanced.

The game engine is extensible and customizable for 3D video games, but only the type: shooter in first person and third person.

## 4 Recommendations

XNA not only can be used in video game development, even a framework for this purpose, since this can lead to many other products as multimedia systems, mobile applications, etc.. A market in the country is not very exploited.

Until the writing this present document, is booming devices with support for Windows Phone 7, XNA implementations for silverlight applications, hardware devices like tablet PC, touch screen applications, among others (early 2012), with the necessary knowledge on this framework would be developed not only PC also for the platforms mentioned, with this into on a new market with great economic benefits.

Develop applications using XNA requires a degree of skill in programming and knowledge fairly advanced programming language, in this case C #, skills that are able to acquire during system development, in the begin the project the knowledge was basic, known the language but no all potential, unfamiliarity with the development of entertainment systems, so it is advisable to test new technology development to increase and improve knowledge.

## References

- [1] Curtis Bennett, A Simple Introduction to Game Programming With C# and XNA 3.1, First Edition: 2009, Estados Unidos de America.
- [2] Jerry Lee Ford, Jr. XNA 3.1 Game Development for Teens, Course Technology PTR A part of engage Learning, Estados Unidos de America.
- [3] David. M Bourg, Physics for Game Developers, Primera edicion, O'Reilly, Estados Unidos de America, Enero 2002.
- [4] Aaron Reed, Learning XNA 3.0, Primera edición, O'Reilly, Estados Unidos de América, Noviembre 2008.
- [5] Aaron Reed, XNA 3.0 Unleashed, Primera edición, O'Reilly, Estados Unidos de América, Enero 2009.
- [6] Rob Miles, Introduction to Programming Through Game Development Using Microsoft XNA Game, Primera edición 2010, Microsoft Press, Estados Unidos de América.
- [7] Rob Miles, C# Development, Edición 1.1, University of Hull, Estados Unidos de América, Octubre 2009.
- [8] Riemer Grootjans, XNA 3.0 Game Programming Recipes, Primera edición 2009, Apress, Estados Unidos de América.
- [9] Alexandre Santos Lobão, Bruno Evangelista, José Antonio Leal de Farias, and Riemer Grootjans, Beginning XNA 3.0 Game Programming From Novice to Professional, Primera edición 2009, Apress, Estados Unidos de América.