

ANALISIS DE LA PROGRAMACIÓN VISUAL

(PROGRAMACIÓN VISUAL)

GRUPO: 4322-IS

INGENIERIA EN SOFTWARE

NOMBRE DEL PROFESOR: EMMANUEL TORRES SERVIN

ALUMN@S: MARQUEZ RODRIGUEZ JOCELYN

EDRIC YAMIL SANCHEZ AVILA

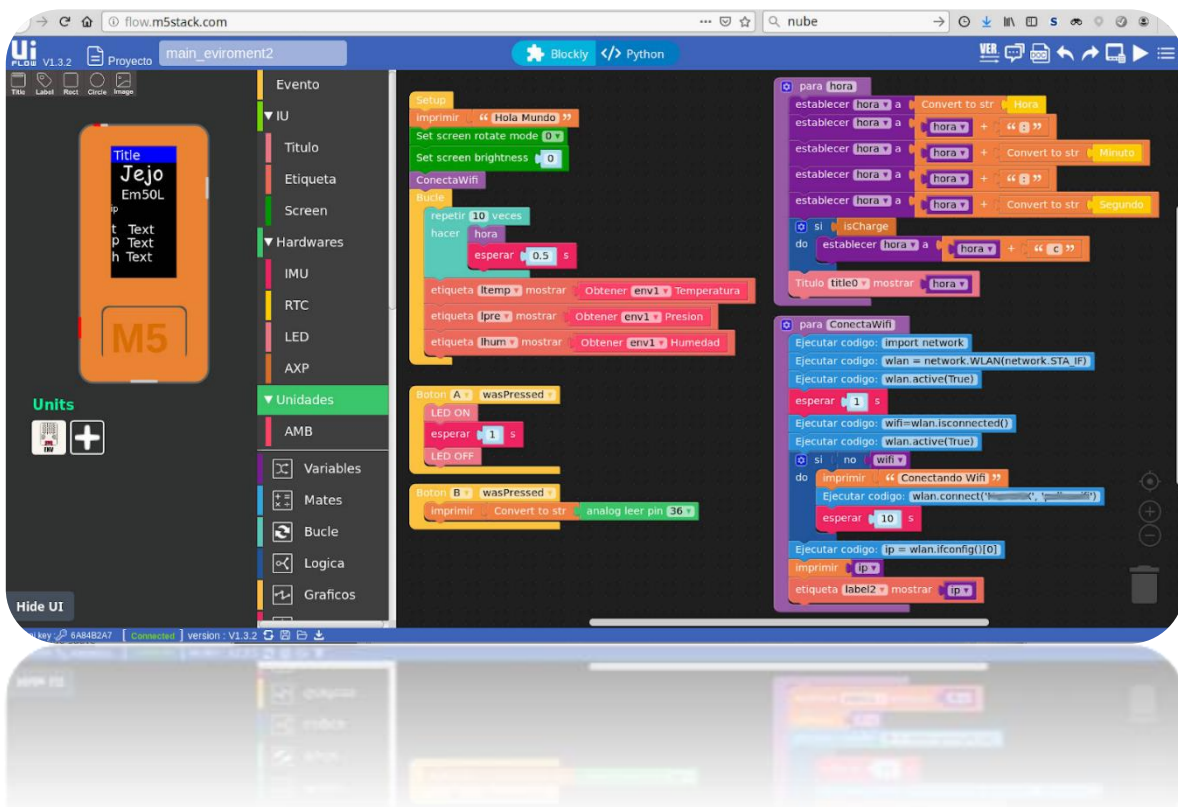
JONATHAN GAMEZ LEDO

INTRODUCCIÓN

En este proyecto se darán a conocer unos cuantos conceptos importantes que nos ayudan sobre este tema y podremos darnos cuenta que gran parte de los lenguajes de programación visual permiten a los usuarios crear programas mediante la manipulación de elementos gráficos, en lugar de especificarlos exclusivamente de manera textual, en donde se pueden utilizar componentes gráficos como, iconos, botones, y símbolos de codificación.

De la misma manera se notará que la programación visual es una forma perfecta de introducir a los usuarios en la codificación y la programación.

Ya que el desarrollo de software moderno como herramienta de programación visual es una solución amigable para los usuarios que no son expertos en codificación.



Conceptos de programación orientada a objetos.

Uno de los casos es el tema de la programación orientada a objetos, la cual se basa en la creación de un modelo del problema de destino en sus programas.

La programación orientada a objetos disminuye los errores y promociona la reutilización del código.

Bien se dice que la programación Orientada a Objetos (POO) es un paradigma de programación, es decir, un modelo o un estilo de programación que nos da unas guías sobre cómo trabajar con él.



Se basa en el concepto de clases y objetos.

Este tipo de programación se utiliza para estructurar un programa de software en piezas simples y reutilizables de planos de código (clases) para crear instancias individuales de objetos.

Características y aplicaciones de eventos.

La programación dirigida por eventos, es un paradigma de programación en el que tanto la estructura como la ejecución de los programas van determinados por los sucesos que ocurran en el sistema, definidos por el usuario o que ellos mismos provoquen.

De la misma manera es la base de lo que llamamos interfaz de usuario, aunque puede emplearse también para desarrollar interfaces entre componentes de Software o módulos del núcleo.

Dependencia de eventos

El flujo del programa viene dado por eventos que pueden ser acciones del usuario, mensajes de otros programas, etc., separando la lógica de procesamiento de los eventos del resto del código de un programa, contrastando así con el procesamiento por lotes.

Orientada al servicio

Se utiliza para escribir programas diseñados para el servicio sin ralentizar la computadora, ya que la orientación al servicio solo consume poco poder de procesamiento.

Eventos

Es una condición que surge durante la ejecución de un programa y que requiere alguna acción por parte del sistema.

Controlador de eventos

Es una determinada unidad del programa que se activa para reaccionar ante un evento.

Es decir, es un tipo de función o método que ejecuta una acción específica cuando se activa un evento determinado.

Funciones de activación

Son funciones que deciden qué código ejecutar cuando se produce un evento específico.

Se utilizan para seleccionar qué controlador de eventos emplear al producirse un evento.

Tiempo controlado

Es un código específico que se ejecuta en un momento determinado.

Características de componentes y métodos visuales y no visuales.

Un componente es visual cuando tiene una representación gráfica en tiempo de diseño y ejecución (botones, barras de scroll, cuadros de edición, etc.), y se dice no visual en caso contrario (temporizadores, cuadros de diálogo no visibles en la fase de diseño, etc).

Procesos de desarrollo visual en proyectos distribuidos y de escritorio.

Un proceso de desarrollo de software es la descripción de una secuencia de actividades que deben ser seguida por un equipo de trabajadores para generar un conjunto coherente de productos, uno de los cuales es el programa del sistema deseado.

Específicamente un proceso de desarrollo debe conseguir:

- Proporcionar una plantilla de desarrollo del proyecto en el que quede definido lo que cada trabajador que interviene debe realizar y los productos que debe generar a lo largo de él.
- Mejorar la calidad del producto que se genera en función de: Disminuir el número de defectos que se producen y que deben ser corregidos.
- Disminuir la severidad de los defectos residuales que al final pueden permanecer en el producto final.
- Mejorar la reusabilidad, de forma que gran parte del trabajo que se realiza pueda ser reutilizado en próximos proyectos.
- Mejorar la estabilidad del proceso de forma que se minimicen las reelaboraciones del producto.
- Generar un producto que sea de fácil mantenimiento posterior.
- Mejorar la predictibilidad del proyecto en función de: La cantidad de esfuerzo humano y de recursos que requiera.
- Disminuir los plazos de desarrollo y llegada al mercado.
- Generar a lo largo del desarrollo de la información adecuada y diferenciada para que los diferentes responsables del proyecto puedan hacer su seguimiento de forma efectiva.

Herramientas y lenguajes de programación visual.

En la programación visual, los elementos del lenguaje de programación están disponibles en forma de bloques diseñados de manera gráfica, por lo que también se la llama programación gráfica.

Los cuales se utilizan como base:

- Python. Este es un lenguaje de programación dinámico y de alto nivel orientado a objetos, que a menudo se usa para el desarrollo rápido de aplicaciones.
- Visual Basic. Este lenguaje de programación orientado a objetos es moderno, muy fácil de aprender y está dirigido a la plataforma.
- C++
- F#

CONCLUSIÓN

Como conclusión nos podemos dar cuenta que el entender la programación visual nos ayuda a mejorar nuestra capacidad de comprender fácilmente la estructura de lenguajes de programación.

REFERENCIAS:

Sáez López, J. M., & Cózar Gutiérrez, R. (2017). Programación visual por bloques en Educación Primaria: Aprendiendo y creando contenidos en Ciencias Sociales. *Revista Complutense de Educación*, 28(2), 409-426.

Duilio, M. C. E., Jorge, A. N. C., Walter, C. C., & Adan, N. Z. W. Programación li-CC67-201601.

Eduardo, B. G. W., Jorge, A. N. C., & Adan, N. Z. W. Programación li-CC67-201502.

Luna, F., Millahual, C. P., & lacono, M. (2018). *PROGRAMACION WEB Full Stack 12-Sitios multiplataforma con Bootstrap: Desarrollo frontend y backend-Curso visual y práctico* (Vol. 12). RedUsers.

DESARROLLO DE APLICACIONES CON PROGRAMACIÓN VISUAL



EVENTOS VISUALES

ES UNA SEÑAL QUE COMUNICA A UNA APLICACIÓN QUE HA SUCEDIDO ALGO IMPORTANTE.

COMPONENTES Y HERRAMIENTAS VISUALES

ENTORNO INTEGRADO DE DESARROLLO QUE PERMITE LA PROGRAMACIÓN ORIENTADA A OBJETOS (POO) CONJUNTAMENTE CON EL SISTEMA DE DESARROLLO SDK

PROPIEDADES DE LOS COMPONENTES VISUALES

Botones, barras de scroll, cuadros de edición, etc.

FUNCIONALIDAD DE LOS EVENTOS VISUALES

Un evento es una señal que comunica a una aplicación que ha sucedido algo importante.

PROCESO DE PROGRAMACIÓN VISUAL

Entre los elementos declarados se incluyen variables, constantes, enumeraciones, clases, estructuras, módulos, interfaces, procedimientos, parámetros de procedimientos, valores devueltos de función, referencias a procedimientos externos, operadores, propiedades, eventos y delegados.

Diseño de programación visual

INTERFACES



PROGRAMACIÓN VISUA



COMPONENTES



PROCESO



INTERACTIVIDAD



COMPONENTES



PROCESOS



MAQUETADO



CONSTRUCCIÓN



MULTIMEDIA



DISEÑO



CONTENIDOS

CONCEPTO Y DESARROLLO DE PROTOTIPOS DE VIDEOJUEGOS

(PROGRAMACIÓN VISUAL)

GRUPO: 4322-IS
INGENIERIA EN SOFTWARE

NOMBRE DEL PROFESOR: EMMANUEL TORRES SERVIN

ALUMN@S: MARQUEZ RODRIGUEZ JOCELYN

EDRIC YAMIL SANCHEZ AVILA

IRVING JONATHAN GAMEZ LEDO

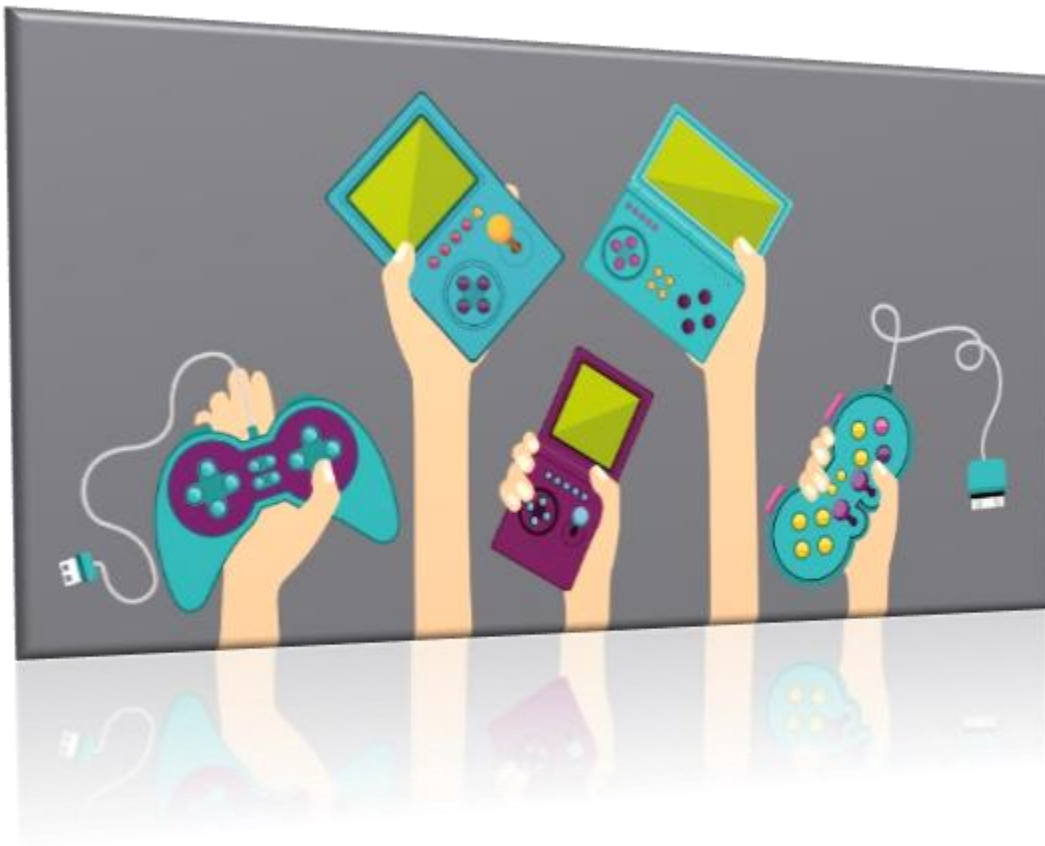
INTRODUCCIÓN

En este proyecto desarrollaremos una investigación la cual demuestra conceptos de desarrollo de videojuegos.

Los videojuegos son un tipo de juego particular, una de esas actividades más o menos estructuradas que creamos para divertirnos.

Lo caracterizamos por utilizar aplicaciones software con gran capacidad multimedia y de interacción gracias a los sistemas electrónicos o informáticos sobre los que se implementan.

Los videojuegos son programas informáticos o software cuya principal función es constituir un medio de entretenimiento para el o los usuarios.



Conceptos y tipos de game designer, storyboard.

Storyboard: El Storyboard no es otra cosa más que la realización de una secuencia de dibujos realizados en una plantilla, acompañados de textos breves que definen la estructura de la historia.

Cumplen con un proceso imprescindible en la fase de preproducción para el desarrollo de Videojuegos que permite previsualizar el resultado final de la producción.

es muy similar al que se emplea en otros sectores como publicidad, cine o televisión.

Su objetivo principal es hacer el videojuego inmersivo, estableciendo la interacción que el jugador tendrá con los diferentes niveles, objetivos y fases del juego.

- **En el Diseño del juego:** Se establece una plantilla principal donde se refleja en las diferentes cuadrículas la estructura base del juego, la vista previa de todos los niveles que conformarán el juego.
- **En el Diseño de niveles:** A partir del storyboard creado anteriormente, creamos un Storyboard para cada uno de los niveles del juego por separado detallando las diferentes acciones, escenas y objetivos del juego. Su función es separar y distinguir cada nivel.
- **En el Diseño de escenas, objetos y personajes:** Se realiza el Storyboard de los personajes, objetos y entornos del juego para describir cómo interactúan entre sí. El Storyboard de un personaje es muy importante, ya que los jugadores serán los que interactuarán con estos personajes durante el juego. Se trata de una forma muy útil de previsualizar y estructurar el contenido de un juego y, se puede realizar con técnicas de dibujo tradicionales o digitales.



TIPOS:

Tradicional: Pueden ser en blanco y negro, a color, bocetos e incluso ilustraciones más elaboradas, todo en función del artista y las necesidades de la producción.

Digital: Mediante softwares que disponen de herramientas para la creación de Storyboards, permite planificar la animación 2D y la creación de recursos para juegos.

Game Desinger:

Game Designer o diseñador de videojuegos es el profesional encargado de diseñar todos los elementos que componen el juego, desde el concepto, las mecánicas o los niveles, entre otros.

Los diseñadores de videojuegos o Game Designers tienen un rol muy importante en el proceso de creación y deben ser conocedores de las tendencias en videojuegos para poder construir un producto atractivo para los consumidores.

TIPOS:

- **Lead designer.** Suele ser quien tiene la idea inicial o en su conjunto. Supervisa la actividad de todos los equipos, comprobando que trabajan en sintonía.
- **Game writer.** Es el autor de la historia, la persona que escribe el guión.
- **Level designer.** Se encarga del diseño de los mapas, fijando su duración, la dificultad o la ubicación y recorrido de los enemigos.
- **System designer.** Diseña las reglas del juego.

Tipos y características de motores de videojuegos y lenguajes de videojuegos

Un motor de videojuego es un término que hace referencia a una serie de librerías de programación que permiten el diseño, la creación y la representación de un videojuego.

De igual manera se habla de el aspecto más destacado a la hora de elegir un motor de videojuegos entre todos los disponibles que hay en el mercado algunas son las capacidades gráficas, ya que son las encargadas de mostrar las imágenes 2D y 3D en pantalla, así como calcular algunos aspectos como los polígonos, la iluminación, las texturas entre otras, Otras características para tener en cuenta a la hora de la elección son la facilidad de aprender a usar el motor de videojuegos y la facilidad para exportar el juego a diferentes plataformas.

El motor de físicas

El motor de físicas es el que hace posible aplicar aproximaciones físicas a los videojuegos para que tengan una sensación más realista en la interacción de los objetos con el entorno.

El motor de sonido

Los sonidos y la banda sonora de un videojuego es también una parte muy importante. El motor de sonidos es el encargado de cargar pistas, modificar su tasa de bits, quitarlas de reproducción, sincronizarlas entre otras cosas.

El scripting

Todos los motores de videojuegos tienen un lenguaje de programación que permite implementar el funcionamiento de los personajes y objetos que forman parte del videojuego.

Dentro de las diferentes opciones de motores de videojuegos podemos distinguirlos en populares y motores propietarios o privados que son los creados por empresas importante de videojuegos para diseñar sus títulos más populares.

Los motores populares más utilizados y que más posibilidades dan al desarrollador son:

- **Unreal Engine:** Fue creado por Epic Games en 1998. En 2012 se presentó Unreal Engine 4, una nueva versión del motor. Entre las empresas que lo utilizan se encuentran Electronic Arts y Ubisoft. Utiliza el lenguaje de programación C++.
- **Unity 3D:** Se trata de una de las innovaciones más importantes creadas por la comunidad científica y de videojuegos y permite jugar a complejos videojuegos en 3D sin necesidad de instalarlos en el ordenador.

Los motores propietarios o profesionales más conocidos por los juegos que han sido diseñados con ellos:

- **Frostbite Engine:** Este motor para videojuegos creado por Digital Illusions CE se utiliza para crear videojuegos de acción en primera persona.
- **Decima Engine:** Alberga herramientas y características para crear inteligencia artificial, física, lógica y mundos en el desarrollo, así como compatibilidad con 4K y HDR.
- **Luminous Studio:** Es un motor de videojuegos multiplataforma desarrollado y usado internamente por Square Enix. Con este motor se desarrolla el juego Final Fantasy.

Metodologías de desarrollo de videojuegos.

Ser diseñador de videojuegos es ser una persona capaz de planificar, crear y producir proyectos orientados a la industria del videojuego, desde su planteamiento y concepción hasta el desarrollo visual del mismo.

Siempre ha sido una profesión demandada pero actualmente y con el *boom* digital esta profesión se ha convertido en imprescindible dentro del sector del entretenimiento.

metodologías utilizadas para el desarrollo de videojuegos siguen principios ágiles por ser iterativas e incrementales, tener interacción frecuente con el cliente y ser flexibles ante los requerimientos cambiantes.

Otra característica es que las decisiones se toman en base a la experiencia, sin existir un proceso definido ni técnicas específicas a seguir.

Existen siete fases importantes para el desarrollo de videojuegos

- Fase de Concepción.
- Fase de Diseño.
- Fase de Planificación.
- Fase de Producción.
- Fase de Pruebas.
- Fase de Mantenimiento

Proceso de diseño de interfaces de videojuegos en 2d y 3d.

La interfaz de usuario en un videojuego es el punto de interacción entre el jugador y el juego.

Su objetivo fundamental es el de brindar la información necesaria para que el usuario pueda hacer todo lo que el juego le propone de manera totalmente fluida.

Para poder diseñar una interfaz de videojuego debemos tener en cuenta:

- **Entorno/Plataforma.** Lo primero a tener en cuenta es dónde se va a jugar el juego que estás diseñando, se debe tener en cuenta las posibilidades y limitaciones que te ofrece la plataforma.

- **Contenido.** Un buen diseño de UI proporciona al jugador toda la información necesaria para que pueda interactuar con el juego y que todo sea fluido.
- **Diseño Visual.** Debes definir el estilo de arte.
- **Arquitectura de la información.** Definir qué elementos son de mayor o menor importancia para el usuario y organizarlos de tal forma que todo resulte en un diseño de interfaz coherente y relevante.

Concepto, tipos y características de los motores de videojuegos.

Para crear un videojuego normalmente se necesita desarrollar, antes de crear el juego en sí, un conjunto de aspectos computacionales que definan cuáles serán los comportamientos de cada uno de los elementos dentro del juego, cómo se verán y cómo reaccionarán ante las diferentes decisiones que el jugador decida de qué forma afectan al objeto en cuestión.

Este es un proceso largo que puede tomar el mismo o mayor tiempo que crear el videojuego, y se enfoca principalmente en la programación, debido a que su desarrollo es principalmente técnico y conlleva una metodología rigurosa propia de desarrollos de software clásico.

Es a este conjunto de estándares (“núcleo” del videojuego) lo que comúnmente se denomina el “motor”.

Tipos.

Motor de Entrada/Salida.

Existen pocos APIs de amplia utilización que faciliten la creación de motores de entrada/salida de libre utilización, siendo DirectInput uno de los estándares más usados profesionalmente, debido a su capacidad de adquirir datos de entrada desde los principales puertos del computador.

Motor de física.

El motor de física se encarga de que el juego tenga leyes de la naturaleza semejantes a las nuestras, por ejemplo, la gravedad, las colisiones elásticas e inelásticas, la fricción, el movimiento de objetos debido al viento (incluyendo al mar), etc.

Motor de audio.

Es el encargado de la parte sonora del videojuego, dándole un sentimiento de realismo mayor al permitir realimentación no solo gráfica sino auditiva. Existen distintas APIs que facilitan la conexión entre los dispositivos de hardware y la programación de alto nivel, las más usadas son OpenAL y DirectX Audio. OpenAL es un estandar multiplataforma utilizado en videojuegos y otras aplicaciones auditivas.

Motor de renderización.

Se encarga de la parte grafica del videojuego.

Es código mediante el cual se visualizan los elementos en la pantalla.

Para realizar esta función existen dos apis estándares en el desarrollo de aplicaciones graficas: opengl (open graphic library) y Direct3D.

Integración de motores de videojuegos con programación visual de acuerdo con los requerimientos del videojuego.

La metodología de creación del motor se fundamentó en el ciclo de vida en espiral, realizando reiteradamente un ciclo de aprendizaje, implementación, pruebas e implantación en el proyecto macro.

De manera general se investigó teoría adecuada sobre los fundamentos principales de cada motor, luego se elaboró la estructura del motor en general, priorizando la modularidad y facilidad de utilización.

Esta estructura sería implementada posteriormente en las subdivisiones de acuerdo con los módulos que se crearon, y la realización de una demostración de cada uno estos se hicieron indispensable.

En una iteración posterior, se realizó la integración de todos los módulos, asegurando la facilidad de utilización tanto de los módulos por separado como su efectividad para crear aplicaciones multimedia utilizando todos los módulos en conjunto.

Este uso también es interesante desde el punto de vista de los clientes ya que pueden ver lo que van a adquirir, se les puede mostrar los productos en sus instalaciones y, por lo tanto, entienden mejor lo que compran por lo que tienen gran aplicación enfocadas a marketing y ventas.

La gran ventaja es que permiten entender a través de la tecnología fenómenos complejos y presentárselos a personas no expertas ya que la Realidad Virtual está ayudando en el proceso de visualización de los datos.

Transición narrativa y lenguaje visual de videojuegos.

La dinámica propia del videojuego propone la interacción con un universo de carácter simbólico pleno de relaciones con objetos y sujetos útiles para el aprendizaje.

La interacción es además apoyada por el desarrollo de procesos de inmersión que facilitan la pertenencia a esos espacios recreados y por el desarrollo de una narrativa particular alternativa a otras formas de representar la realidad.

Un lenguaje de programación de videojuegos, secuencias de comandos, o código, es el sistema que emplean los programadores de videojuegos para dar forma a la configuración y a las mecánicas de un juego.

Para realizar poder realizar la programación pertinente han podido sobre salir algunos lenguajes, los más populares si no es que los más utilizado para programar esta cuestión son c++, c#, javascrip, lua, Python

La forma en que se codifica un juego depende de muchos factores, como, por ejemplo, del tipo de proyecto que se esté desarrollando.

No es lo mismo hacer un juego para consolas que para ordenador.



Explicación del proceso de desarrollo de videojuegos acorde a los elementos de programación visual.

El desarrollo de videojuegos es el proceso de creación de un videojuego, desde el concepto inicial hasta el videojuego en su versión final.

El proceso es similar al desarrollo de software general, aunque difiere en muchos aportes creativos (música, historia, diseño de personajes, niveles, etc.). El desarrollo también varía según la plataforma de destino (PC, dispositivos móviles, consolas), el género (estrategia en tiempo real, RPG, aventura gráfica, plataformas, etc.) y el formato de visualización (2D, 2.5D, 3D).

Diseño.

En esta etapa se detallan todos los elementos que componen el juego, lo que da a todos los miembros del equipo de desarrollo una idea clara de lo que son. Entre estos artículos tenemos:

Historia: cómo los personajes del juego se desarrollarán y representarán la historia del mundo (o de un planeta en particular).

Escenario: El proceso comienza con una reunión de todo el equipo de desarrollo para que todo el equipo tenga la oportunidad de aportar sus ideas o sugerencias al proyecto. A partir de ahí, el equipo de historia trabaja para obtener un esquema que refleje los objetivos del juego, las partes en las que se dividirá, el contexto en el que se desarrollará la acción, quiénes y cómo serán los personajes.

Arte conceptual: creó el aspecto general del juego. Durante esta etapa, un grupo de artistas es responsable de dibujar o renderizar personajes, escenarios, criaturas, elementos, etc.

sonido: detallada descripción de todos los elementos sonoros que el juego necesita para su realización.

Voces, sonidos ambientales, efectos sonoros y música.

Mecánica de juego: es la especificación del funcionamiento general del juego. Es dependiente del género y señala la forma en que los diferentes entes virtuales interactuarán dentro del juego,

Diseño de programación: describe la manera en que el videojuego será implementado en una máquina real (PC, consola, móviles, etc) mediante un cierto lenguaje de programación y siguiendo una determinada metodología

Preproducción.

Durante la etapa de preproducción, los proyectos se asignan a un pequeño equipo para probar la viabilidad de una idea.

El equipo trabajará para crear los niveles o entornos del juego lo más cerca posible del producto final.

La preproducción es una de las partes más críticas del proceso, ya que determinará la viabilidad del juego.

Producción.

Programación: la mayoría de los juegos están escritos en C++ porque es un lenguaje de nivel medio que brinda acceso rápido a los componentes de hardware de una computadora o consola de juegos, haciéndolo más accesible.

Descripción: los juegos 2D deben ser ilustrados por artistas experimentados cuyo trabajo tenga en cuenta las limitaciones técnicas del hardware en el que se ejecuta el juego, que incluyen: la cantidad de colores disponibles, el tamaño de los sprites, la resolución final de los sprites y el formato usado. Los artistas 2D también son responsables de animar el juego.

Interfaz: Esta es la apariencia de los elementos GUI y HUD a través de los cuales el usuario interactuará con el juego.

Modelado y animación 3D: los artistas utilizan herramientas comerciales de animación y modelado 3D como 3DS Max, Maya, XSI/Softimage, Blender

Sonido: los ingenieros de sonido son responsables de crear sonidos para cada objeto o personaje del juego.

Pueden crear sonidos desde cero o usar sonidos del entorno y modificarlos según sea necesario.

Pruebas.

Al igual que en otros tipos de software, los videojuegos deben pasar en su desarrollo por una etapa donde se corrigen los errores inherentes al proceso de programación y se asegura su funcionalidad.

Pruebas físicas: Estas pruebas son realizadas por los diseñadores y programadores del juego. Crea prototipos para simular posibles eventos en el juego.

Los prototipos físicos pueden usar papel y lápiz, fichas e incluso actuar. Según los resultados de estas pruebas, el equilibrio del juego se puede aproximar mejor y pueden evitar problemas de programación

Prueba alfa: realizada por un pequeño grupo de personas involucradas previamente en el desarrollo, que pueden incluir artistas, programadores, coordinadores, etc.

Pruebas Beta: Estas pruebas son realizadas por equipos de jugadores externos que son contratados para esto o son parte del proyecto (grupo QA).

A partir de estas pruebas, el juego debería tener la menor cantidad posible de fallas menores y ninguna falla moderada

CONCLUSIÓN

Como conclusión tenemos que la creación de prototipos en el ámbito del diseño de videojuegos es una práctica muy común.

Es recomendable para evitar y mitigar los riesgos de fases posteriores, una manera sencilla de comprobar nuestras capacidades y estudiar la viabilidad a la hora implementar de nuestras ideas.

El prototipado se consagra de esta manera como la mejor manera de experimentar en una situación de seguridad de forma sencilla y barata, que nos permitirá aprender y mitigar riesgos para poder seguir avanzando en nuestros proyectos partiendo de una base sólida.



REFERENCIAS:

Acerenza, N., Coppes, A., Mesa, G., Viera, A., Fernández, E., Lorenzo, T., & Vallespir, D. (2009). Una MetodologíaAgil para Desarrollo de Videojuegos.

Triviño Monje, J. P. (2022). Desarrollo del prototipo de videojuego Come On! Test!.

Baquero Rojas, D. F., & Vasco Escobar, D. E. Desarrollo de un prototipo de Videojuego Hyper-Casual para dispositivos móviles.

González, C. S. G., & Izquierdo, F. B. (2011). Videojuegos educativos sociales en el aula. *Revista ICONO 14. Revista científica de Comunicación y Tecnologías emergentes*, 9(2), 59-83.

Neyra Celis, D., & Vivar Álvarez, K. F. (2016). Diseño y desarrollo del prototipo de videojuego educativo" Geniux" para reforzar el aprendizaje en el área de matemática en el 2do grado de primaria.

Muñoz Pérez, J. (2021). QUEST FOR HARMONY: PROTOTIPO DE VIDEOJUEGO PARAEL ANÁLISIS DE LAS ACCIONES DEL JUGADOR. *Licenciatura en Diseño Digital de Medios Interactivos*.

Ramírez Caiza, K. S., & Velasco Galarraga, S. G. (2022). *Desarrollo de un juego prototipo utilizando el motor Unreal Engine para videojuegos en 3D* (Bachelor's thesis).