



FACULTAD DE INGENIERÍA UNAM

**PROFESOR:
ING. JOSE ROQUE RG**

**COURSE:
COMPUTACION GRAFICA IH**

GROUP: 06

**FINAL PROYECT
TECHNICAL MANUAL**

**ALUMNOS:
CRUZ CEDILLO DANIEL ALEJANDRO
CHAVEZ FLORES GIOVANNI**

**SEMESTRE:
2023-2**

**FECHA DE ENTREGA:
15/06/2023**

Observaciones:

TECHNICAL MANUAL

INDICE

- OBJECTIVE
- INTRODUCTION
- TECHNICAL REQUIREMENTS
- VISUAL STUDIO
- INSTALLATION AND CONFIGURATION IRRKLANG
- INSTALLATION AND CONFIGURATION KEYS AND CONTROLS
- STAGE MAP
- MODELS
- ANIMATIONS
 - COMPLEX SONIC
 - SIMPLE FREDDY
 - SIMPLE CHICA
 - COMPLEX EGGMAN
 - COMPLEX CHEF
 - SIMPLE ANIMATION
 - RINGS KEYFRAMES
- LIGHTING
 - DAY AND NIGHT CYCLE
 - AUTOMATIC SPOTLIGHT ON AND OFF
 - SPOTLIGHT ON AND OFF VIA KEYBOARD
- REFERENCES

OBJECTIVE

The purpose of this manual is to provide an understanding of the program's operation, animation functionality, activity schedule, and bibliographic sources.

To achieve this, we will seek to implement all the knowledge acquired throughout the course, such as lighting, rendering, animation, among other elements, in OpenGL.

INTRODUCTION

To implement all the knowledge acquired throughout the course, such as lighting, rendering, animation, among other elements, in OpenGL. The purpose of this manual is to provide an understanding of the program's operation, animation functionality, the animations themselves, the activity schedule, and bibliographic sources.

For the development of this project, Visual Studio and the C++ language were used, along with OpenGL libraries and audio libraries.

REQUERIMIENTOS TECNICOS

- HARDWARE
 - PROCESOR: INTEL I5 O SIMILAR
 - RAM: 8 GB RAM
 - MEMORY: 1GB
- SOFTWARE
 - VISUAL STUDIO
 - OPENGL
 - IRRKLANG
 - BLENDER

INSTALLANTION VISUAL STUDIO

Next, we will explain how to install Visual Studio for Windows. To do this, you will need to access an internet browser and type "Visual Studio" in the search bar. Access the Microsoft website to download it.

microsoft.com
https://visualstudio.microsoft.com > ... :

Visual Studio: IDE y Editor de código para desarrolladores de ...

IDE de **Visual Studio** es una plataforma de lanzamiento creativa que puede utilizar para editar, depurar y compilar código y, finalmente, publicar una aplicación.

Once inside, scroll down a bit until you find three download menus. Select the one for Windows and download Visual Studio Community 2022.

Conozca la familia de Visual Studio

Visual Studio | El mejor IDE completo para desarrolladores de .NET y C++ en Windows. Completamente equipado con una buena matriz de herramientas y características para elevar y mejorar todas las etapas del desarrollo de software.

Más información →

Descargar Visual Studio ▾

Community 2022 Professional 2022 Enterprise 2022

¿Viene más? Podemos ayudarle.

Visual Studio para Mac | Un IDE completo nativo de macOS para desarrolladores de .NET. Incluye compatibilidad de primer nivel para el desarrollo de la web, la nube, los móviles y los juegos.

Más información →

Más información sobre activando su licencia

Descargar Visual Studio para Mac

Visual Studio Code | Editor de código fuente independiente que se ejecuta en Windows, macOS y Linux. La elección principal para desarrolladores web y JavaScript, con extensiones para admitir casi cualquier lenguaje de programación.

Más información →

Al usar Visual Studio Code, confirma que acepta la licencia & declaración de privacidad

Descargar Visual Studio Code ▾

Download the file, preferably save it in the Downloads folder. Once the download is complete, double-click the executable file. It will open a screen asking for permissions, to which you should click "Accept".

After that, an installer will open, and you will need to choose to install the C/C++ packages and libraries. Once selected, proceed with the installation.

Once the installation is complete, you will be able to use Visual Studio.

INSTALLATION AND CONFIGURATIONIRRKLANG

irrKlang Installation:

1. To install irrKlang, first download the files from the following link. The 32-bit version is the one that will be used:

<https://www.ambiera.com/irrklang/downloads.html>

After the RAR file is downloaded, open it and navigate to the "include" folder. Copy all the contents of this folder to the "include" folder within your project.

Carpeta de archivos					
ik_ESoundEngineOptions.h	4,183	1,771	Archivo H	29/01/2018 11:...	D262BA9C
ik_ESoundOutputDrivers.h	2,289	944	Archivo H	29/01/2018 11:...	E8BFFACE
ik_EStreamModes.h	829	462	Archivo H	29/01/2018 11:...	2FEEDB66
ik_IAudioRecorder.h	5,185	1,655	Archivo H	29/01/2018 11:...	2851260F
ik_IInputStream.h	1,736	816	Archivo H	29/01/2018 11:...	302A684F
ik_IInputStreamLoader.h	1,242	641	Archivo H	29/01/2018 11:...	6D90072B
ik_IFileFactory.h	1,537	655	Archivo H	29/01/2018 11:...	6B385E2B
ik_I.FileReader.h	1,676	703	Archivo H	29/01/2018 11:...	6A97B46E
ik_IRefCounted.h	4,710	1,298	Archivo H	29/01/2018 11:...	467D4259
ik_irrklangTypes.h	2,978	1,016	Archivo H	29/01/2018 11:...	E45175BE
ik_ISound.h	10,023	3,109	Archivo H	29/01/2018 11:...	85F8F2DC
ik_ISoundDeviceList.h	1,704	740	Archivo H	29/01/2018 11:...	5A7A78A3
ik_ISoundEffectControl.h	14,359	3,381	Archivo H	29/01/2018 11:...	AF51B050
ik_ISoundEngine.h	28,623	6,045	Archivo H	29/01/2018 11:...	F65EC8B4
ik_ISoundMixedOutputReceiver.h	1,827	947	Archivo H	29/01/2018 11:...	BCF84653
ik_ISoundSource.h	9,885	2,609	Archivo H	29/01/2018 11:...	B0C826D3
ik_ISoundStopEventReceiver.h	3,017	1,135	Archivo H	29/01/2018 11:...	6C0D9DB6
ik_IVirtualRefCounted.h	1,306	569	Archivo H	29/01/2018 11:...	61A672B1
ik_SAudioStreamFormat.h	1,713	725	Archivo H	29/01/2018 11:...	02DF959C
ik_vec3d.h	8,382	2,366	Archivo H	29/01/2018 11:...	DC7E884C
irrklang.h	37,883	10,296	Archivo H	30/01/2018 09:...	CB440BE9

Debug	01/06/2023 11:56 a. m.	Carpeta de archivos
<input checked="" type="checkbox"/> include	30/05/2023 10:57 p. m.	Carpeta de archivos
lib	30/05/2023 10:29 p. m.	Carpeta de archivos
Release	01/06/2023 11:54 a. m.	Carpeta de archivos
resources	01/06/2023 11:36 a. m.	Carpeta de archivos
shaders	19/04/2023 01:24 p. m.	Carpeta de archivos
Texturas	30/03/2023 02:09 p. m.	Carpeta de archivos
x64	18/02/2023 03:51 p. m.	Carpeta de archivos

- 2.- In the RAR file, go to "lib/Windows32-visualStudio" and copy the "irrklang.lib" file into the "lib" folder of our project.

Carpeta de archivos					
irrklang.exp	2,082	568	Exports Library File	12/02/2018 08:...	C245B962
irrklang.lib	4,322	780	Object File Library	12/02/2018 08:...	558AFFE8

INSTALACION Y CONFIGURACION IRRKLANG

3. Next, in the RAR file, go to "bin/Windows32-visualStudio" and copy the "irrklang.dll" file into the main project folder.

		Carpeta de archivos			
01.HelloWorld.exe	483,328	152,221	Aplicación	12/02/2018 09:...	6BA45732
02.3DSound.exe	94,208	41,632	Aplicación	12/02/2018 09:...	2FFADD54
03.MemoryPlayback.exe	65,536	30,326	Aplicación	12/02/2018 09:...	B54CC2FD
04.OverrideFileAccess.exe	81,920	34,951	Aplicación	12/02/2018 09:...	330B3519
05.Effects.exe	77,824	35,442	Aplicación	12/02/2018 09:...	88744BAD
06.Recording.exe	73,728	32,533	Aplicación	12/02/2018 09:...	8B64E374
ikpFlac.dll	159,744	77,156	Extensión de la apl...	12/02/2018 08:...	82D93EA3
ikpMP3.dll	163,840	85,197	Extensión de la apl...	12/02/2018 08:...	63F0F2BD
irrklang.dll	536,576	265,382	Extensión de la apl...	12/02/2018 08:...	43A138E9
irrklangPlayer.exe	360,448	164,644	Aplicación	12/02/2018 09:...	6BA6A4FD

4.-Afterward, open Visual Studio and go to properties.

in C/C++>general>Directorios de inclusion adicionales>include;glm;

The screenshot shows the 'Propiedades de configuración' (Properties) window for a project. The left pane lists categories: General, Avanzado, Depuración, Directorios de VC++, C/C++, and General. The 'General' category is selected. The right pane shows the 'Directarios de inclusión adicionales' (Additional Include Directories) property, which contains the value 'include;glm;'.

En Vinculador>Directorios de bibliotecas adicionales>lib

The screenshot shows the 'Vinculador' (Linker) section of the properties window. Under 'General' settings, the 'Directarios de bibliotecas adicionales' (Additional Library Directories) property is set to 'lib'.

in Vinculador>Entrada>Dependencias adicionales>irrklang.lib;

The screenshot shows the 'Entrada' (Input) section of the properties window. Under 'General' settings, the 'Dependencias adicionales' (Additional Dependencies) property is set to 'irrklang.lib;assimp-vc141-mtd.lib;opengl32.lib;glfw3.lib;SD'.

SAVE CHANGES AND CLOSE

INSTALACION Y CONFIGURACION IRRKLANG

5.-Add the irrKlang.h library to the file in order to use it.

```
#include <glm/gtc/matrix_transform.hpp>
#include <glm/gtc/type_ptr.hpp>
#include <time.h>
#include <irrklang.h> //Sonido
```

6.- To make it work, the following steps need to be performed in the main function. First, create a variable for the sound. Then, use an if statement to check if the sound variable is initialized.

Next, assign the sound to the sound variable as a 2D sound, which will be the background sound. After that, set the volume for the sound variable to the desired level during execution.

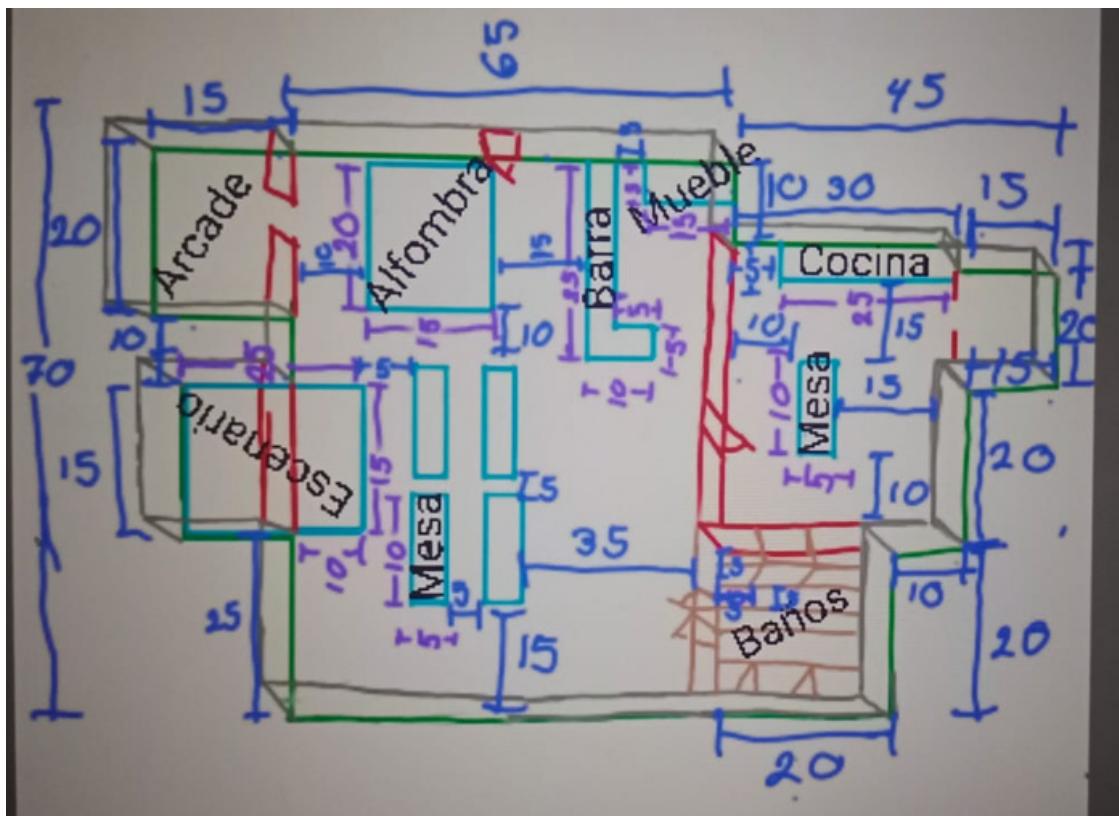
```
irrklang::ISoundEngine* engine = irrklang::createIrrKlangDevice();

if (!engine) {
    printf("Could not startup engine ");
    return 0; //error inciando engine
}

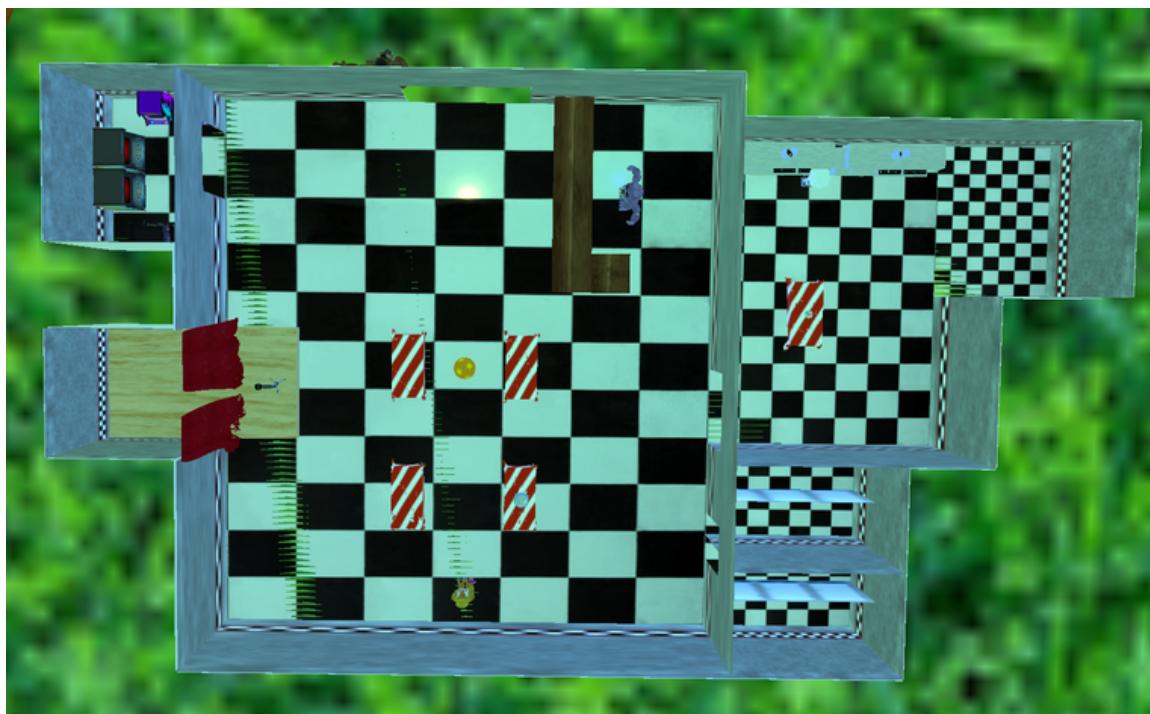
engine->play2D("fnaf.mp3", true);
engine->setSoundVolume(0.4f);
```

STAGE MAP

For the scenario, which in this case is the restaurant, the following map was followed, and it was faithfully replicated as much as possible. However, some changes were made to it in the end, and some unnecessary furniture items were removed.

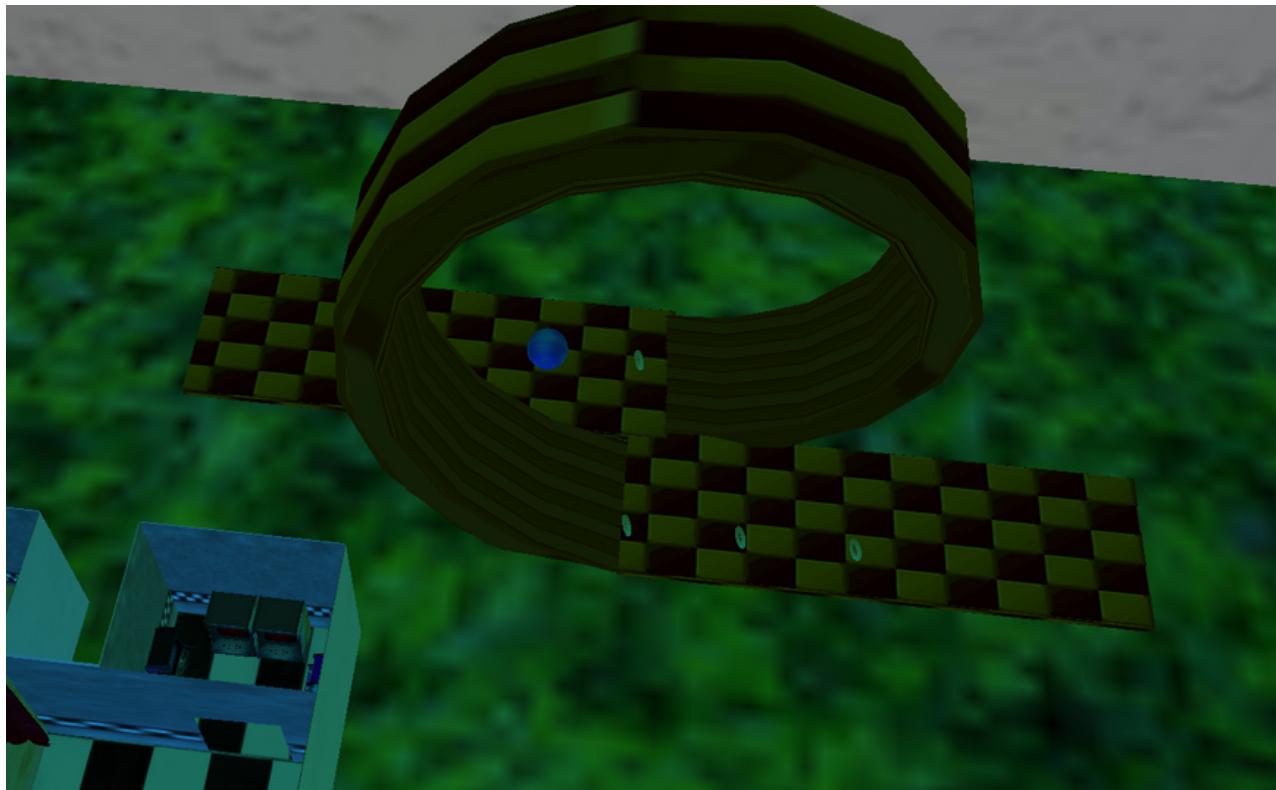


Likewise, an effort was made to preserve the measurements proposed in the plan as closely as possible.



STAGE MAP

External details were also added, which were not initially considered, and efforts were made to incorporate them in a way that wouldn't interfere with the restaurant.



KEYS AND CONTREOLLS

To interact within the program's world, the following keys will be used:

- Key W: Pressing the W key allows the user to move forward.
- Key A: Pressing the A key enables leftward movement.
- Key S: Pressing the S key triggers backward movement.
- Key D: Pressing the D key facilitates rightward movement.
- Mouse: Moves the camera.
- Key P: Activates the Ring animation.
- Key 1: Starts and stops Freddy's animation.
- Key 2: Starts and stops Chica's animation.
- Key 3:
- Key 9: Turns the Arcade light on and off.
- Key P: Activates the keyframe animation of the rings.

MODELS

In the program, there are more than 15 models, and Blender was used as the tool for modeling and utilizing them.

Using models proved to be a much more convenient and faster way to create objects, which greatly aided in decorating the scenario. It also facilitated the animation process. The distribution of the models was as follows:

- Main Hall:
 - Table: For this model, a base model available for free on the Internet was used. The focus was mainly on the tablecloth, which was modified, and legs were added to make it fit into the environment.
 - Microphone: This model was easy to manipulate, as no major changes were needed apart from scaling and placing it in the right position.
 - Cake: A decorative element considered essential in a party hall.
 - Bunny: Added as a decorative element in the scene without any special function.
 - Chica: A character from FNAF (Five Nights at Freddy's) used for one of the animations.



MODELS

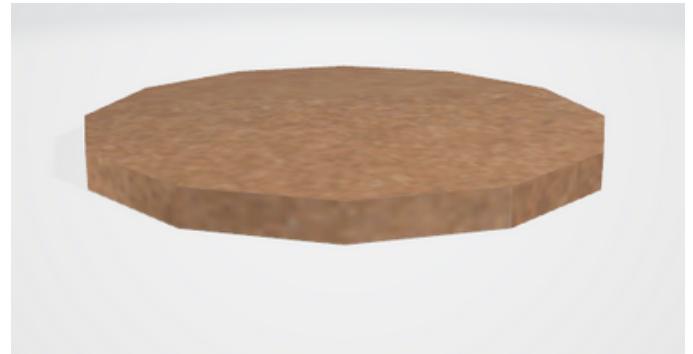
- Arcade Room:
 - Arcade Machine: Three different models of arcade machines were used in this room, but only one of them has added lighting to simulate it being turned on.



MODELS

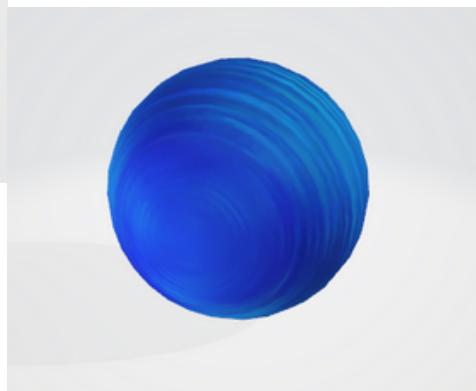
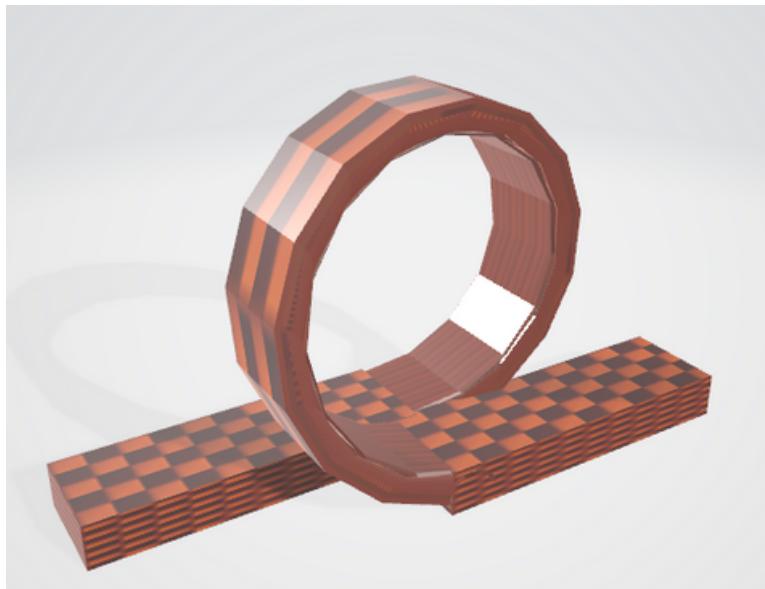
- Kitchen:

- Stove: It was one of the most challenging elements to find as there were few free models available, and many of the ones found didn't fit the theme or were difficult to manipulate for use.
- Chef: This character was included as an essential part of an animation, performing a specific action.
- Meat: This element was used in an animation where it goes through the majority of the movement.
- Frying Pan: This object was extracted from another model using Blender's tools.
- Plate: A decorative element that was also included as part of the animation.



MODELS

- Exterior:
 - Freddy: A model downloaded from Blender, and using Blender's tools, the arm was separated from the rest of the body to give it animations.
 - Eggman: A model added with the purpose of including an animation related to the characters.
 - Sonic: A downloaded model used for animation, chosen by one of the team members.
 - Restaurant: A purely designed element created with Blender by the team members, which was essential for the diorama.
 - Sonic Map: Designed by one of the team members to decorate and give meaning to Sonic's animation.
 - Rings: Decorative part of the Sonic scene to which keyframe animations were added.

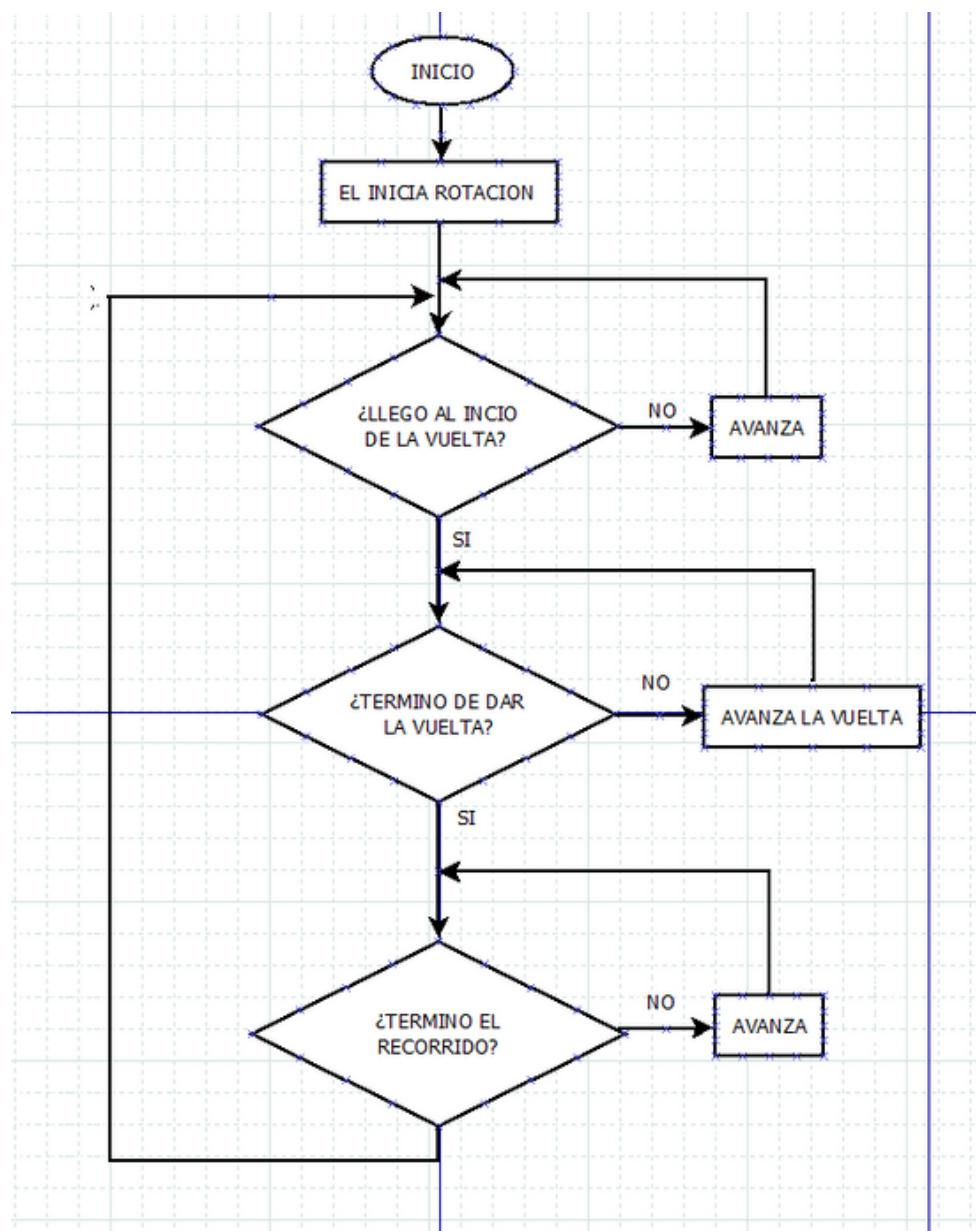


ANIMATIONS

For the animations, we primarily relied on the concept of a state machine, where the animation performed depends on the current state. To achieve this, a switch statement was used in most of the animations to easily switch between different states.

- "SONIC"

For Sonic's animation, we aimed to create a path traversal. This animation is located on the right side of the stage, outside the restaurant. The animation was based on the following flowchart:



ANIMATIONS

This animation had a total of 6 cases. The first 3 cases are responsible for performing the animation in one direction, while the other 3 cases are responsible for performing the animation in another direction.

```
case 0:  
    posys += 0.6f;  
    if(posys >= 150)  
        animsonic = 1;  
    break;  
case 1:  
    //Escala * 3  
    poszs = 95 + ( -95 *cos(incsonic));//altura circ  
    posys = 150 + ( 95 *sin(incsonic));//largo circ  
    incsonic += 0.01;  
  
    posxs -= 0.1;  
    if (incsonic >= 6.5) { //3  
        animsonic = 2;  
    }  
    if (posxs <= -66) {  
        posxs = -66.0f;  
    }  
    break;  
case 2:  
    posys += 0.6f;  
    if (posys >= 350)  
        animsonic = 3;  
    break;  
  
case 3:  
    posys -= 0.6f;  
    if (posys <= 150)  
        animsonic = 4;  
    break;
```

Case 0 is responsible for positioning Sonic where he will start the circular path and where he will end. Case 1 performs the circular route with a small displacement, and case 2 completes the traversal.

This route is repeated in reverse with cases 4, 5, and 6.

```
model = glm::translate(glm::mat4(1.0f), glm::vec3(300.0f, 5.0f, 150.0f));  
model = glm::scale(model, glm::vec3(8.0));  
model = glm::rotate(model, glm::radians(90.0f), glm::vec3(0.0f, 1.0f, 0.0f));  
staticShader.setMat4("model", model);  
mapa.Draw(staticShader);  
  
model = glm::translate(glm::mat4(1.0f), glm::vec3(posxs + 340.0f, poszs + 11.0f, posys));  
model = glm::rotate(model, glm::radians(rotsonic), glm::vec3(1.0f, 0.0f, 0.0f));  
staticShader.setMat4("model", model);  
sonic.Draw(staticShader);
```

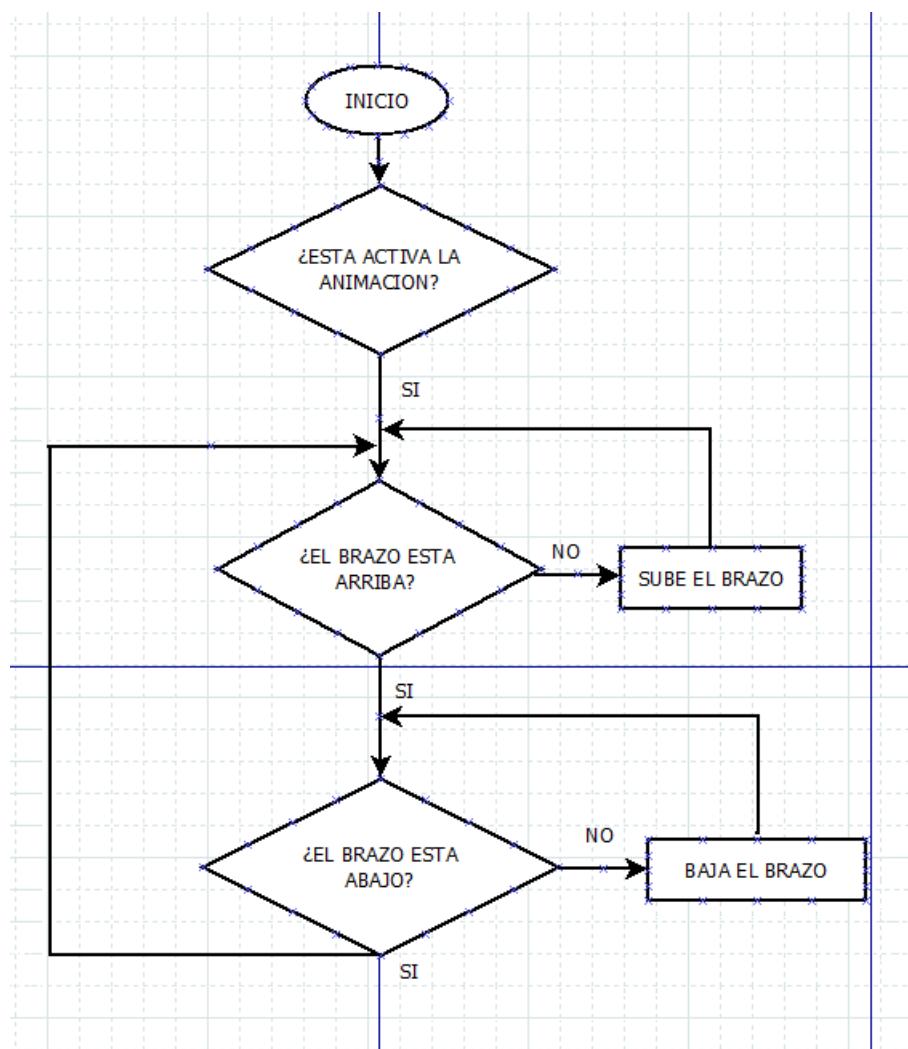
ANIMATIONS

The flowchart is designed in one direction, but in the animation, the traversal was made to go back and forth. To achieve this, sine and cosine functions were used, specifying the center of the circle as the center of the path.

For this purpose, the center was set at (0, -95, 95). Since the rotation is more elliptical, two diameters were used: the first with a length of 95 and the second with a length of 150.

- "FREDDY"

Freddy's animation consists of a greeting. It is located next to the entrance of the restaurant. The following flowchart was used for this animation:



The animation consists of increments and decrements in the rotation of the arm to simulate a greeting.

ANIMATIONS

For Freddy's animation, only 2 cases were used.

```
if (Freddyanim) {
    switch (animFreddy) {
        case 0:
            rotBrazoF += 1.0f;
            if (rotBrazoF >= 45)
                animFreddy = 1;
            break;
        case 1:

            rotBrazoF -= 1.0f;
            if (rotBrazoF <= -45)
                animFreddy = 0;
            break;
    }
}
```

The first case is responsible for lifting the arm, while the second case is responsible for lowering it. This is achieved through rotation. The arm was separated from the body to perform this animation.

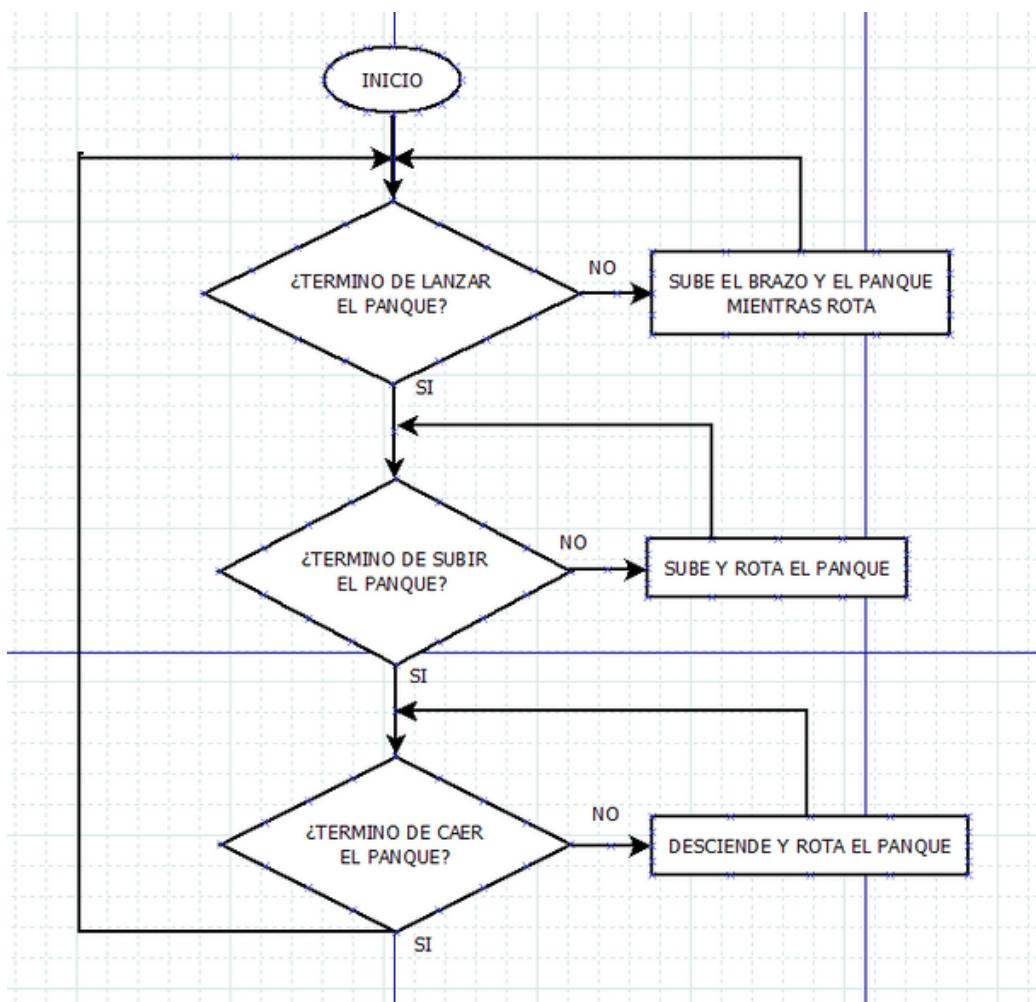
```
model = glm::translate(glm::mat4(1.0f), glm::vec3(47.0f, 34.5f, 48.0f));
model = glm::scale(model, glm::vec3(10.0));
model = glm::rotate(model, glm::radians(rotBrazoF), glm::vec3(0.0f, 0.0f, 1.0f));
staticShader.setMat4("model", model);
FreddyBrazo.Draw(staticShader);
```

Freddy's animation starts playing when the program begins, but it can be stopped by the user and resumed as well.

ANIMATIONS

- "CHICA"

For Chica's animation, the goal was to move her arm and the pancake she holds. The animation aims to simulate Chica throwing the pancake, with the pancake moving up and down while rotating throughout the animation. The following flowchart was followed:



For the animation, it was necessary to perform a rotation of the arm and translate the object while it undergoes a rotation. The most challenging part of this animation is coordinating the rotation animation with the speed at which the object moves up and down.

ANIMATIONS

For Chica's animation, only 3 cases were used. The first case is responsible for raising the hand and the pancake, the second case is for completing the raising of the pancake, and the third case is for lowering the hand and the pancake back to the initial position to repeat the animation.

```
switch (animChica) {
case 0:
    rotBrazoC -= 0.3f;
    poszpanque += 0.1f;
    rotpanque += 2.7;
    if (rotBrazoC <= -20)
        animChica = 1;
    break;
case 1:
    poszpanque += 0.3f;
    if (poszpanque >= 25)
        animChica = 2;
    break;
case 2:
    rotBrazoC += 0.3f;
    rotpanque += 2.7;
    poszpanque -= 0.105f;
    if (rotpanque >= 360)
        rotpanque = 0;
    if (rotBrazoC >= 0)
        animChica = 0;
    break;
}
```

The arm and the pancake are separated from the body and are responsible for the animation.

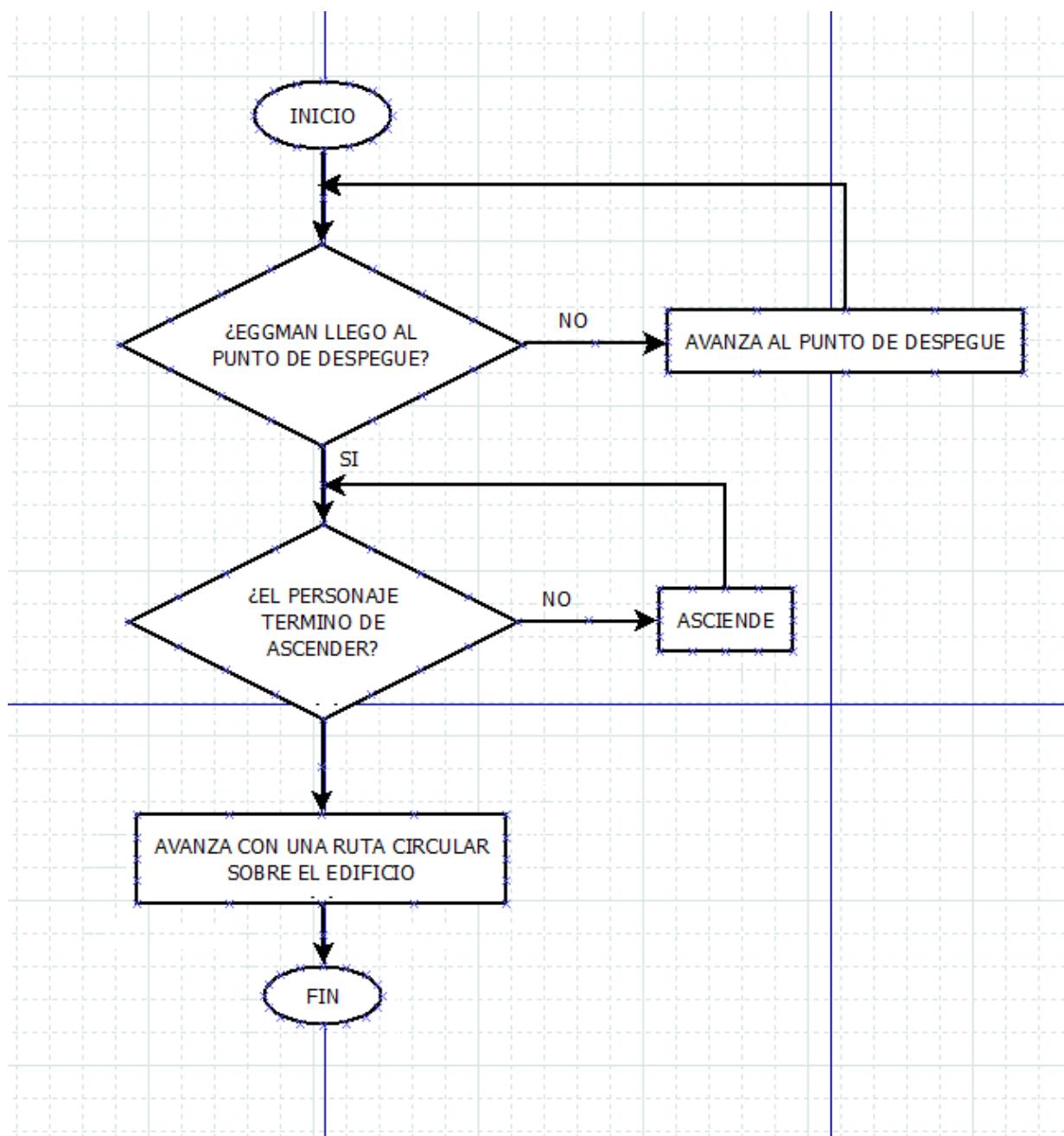
```
model = glm::translate(glm::mat4(1.0f), glm::vec3(-4.5f, 17.0f, -218.5f));
model = glm::scale(model, glm::vec3(0.3));
model = glm::rotate(model, glm::radians(rotBrazoC), glm::vec3(1.0f, 0.0f, 0.0f));
staticShader.setMat4("model", model);
ChicaBrazo.Draw(staticShader);

model = glm::translate(glm::mat4(1.0f), glm::vec3(-4.5f, poszpanque, -212.0f));
model = glm::scale(model, glm::vec3(0.025));
model = glm::rotate(model, glm::radians(rotpanque), glm::vec3(1.0f, 0.0f, 0.0f));
staticShader.setMat4("model", model);
panque.Draw(staticShader);
```

ANIMATIONS

- "EGGMAN"

For Eggman's animation, the goal was to make the character exit the restaurant, rise up, and spin above it. The following flowchart was used for this animation.



For the ascending part of the character, the goal was to make it ascend in a partial circle and then transition to a wider circle.

For the circular path, sine and cosine functions were used, with the center of the restaurant as the center of the circle.

ANIMATIONS

It was one of the most challenging animations to complete because the ascent didn't go as intended, and I had to make several calculations to get it right.

```
case 0:
    eggx += 0.7f;
    if (eggx >= 70)
        animegg = 1;
    break;
case 1:
    rotegg += 0.7f;

    if (rotegg >= 90)
        animegg = 2;
    break;
case 2:
    eggx += 0.7f;
    if (eggx >= 200)
        animegg = 3;
break;
case 3:
    rotegg -= 0.7f;

    if (rotegg <= 0)
        animegg = 4;
break;
case 4:
    eggx -= 0.7f;
    if (eggx <= 40)
        animegg = 5;
    break;
case 5:
    eggz += 0.3f;
    eggx = 200 * cos(egginc);
    eggz += 0.3f;
    eggy = 200 * sin(egginc);
    eggz += 0.3f;
    egginc += 0.01f;
    rotegg -= 0.4f;
    if (eggz >= 100)
        animegg = 6;
    break;
case 6:
    rotegg -= 0.46f;
    eggx = 200 * cos(egginc);
    eggy = 200 * sin(egginc);
    egginc += 0.008f;
    break;
```

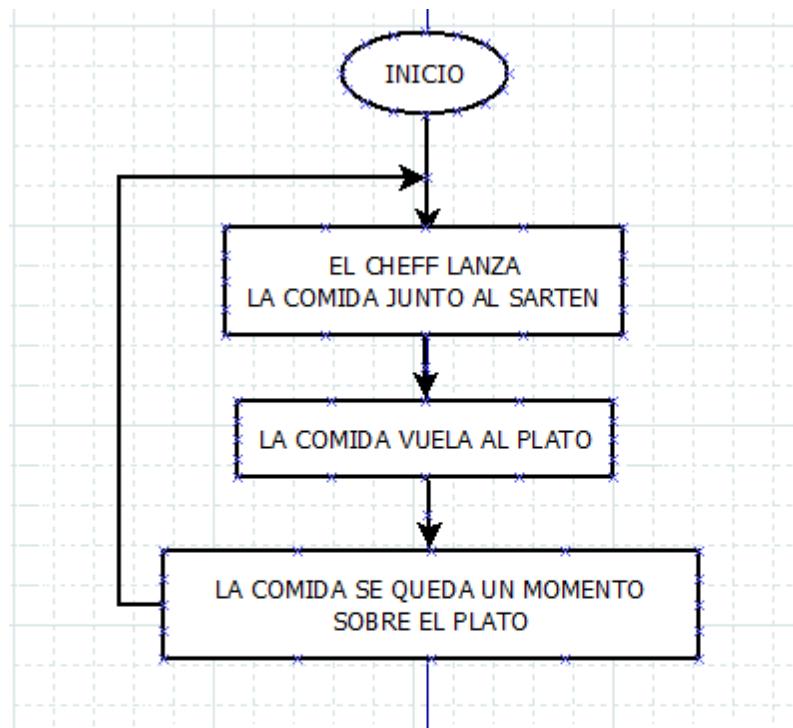
The first 5 states are responsible for positioning the character at the takeoff point since the character initially leaves the restaurant and reaches its takeoff point. After that, in state 5, the character has to take off, which was the most complicated part. Finally, the character stays in the sky, repeating the same animation and circling around the restaurant.

```
model = glm::translate(glm::mat4(1.0f), glm::vec3(eggx, eggz, eggy));
model = glm::scale(model, glm::vec3(3.0));
model = glm::rotate(model, glm::radians(rotegg), glm::vec3(0.0f, 1.0f, 0.0f));
staticShader.setMat4("model", model);
Eggman.Draw(staticShader);
```

ANIMATIONS

- "CHEFF"

For the Chef's animation, we aim to create the animation where he throws a pan with meat, and then the meat flies and lands on a plate on a table. Meanwhile, the pan falls back into the Chef's hands. We based this animation on the following flowchart.



For this animation, several elements needed to be animated. Rotations were used for the arms to simulate the pan throwing motion. The pan itself was given an increment and a rotation to depict the throw. The meat was animated using circular paths to create its trajectory.

The most challenging part of this animation was transitioning between the two circular paths to achieve a seamless and continuous movement. Different increments were applied to create variation in the motion.

ANIMATIONS

It was also one of the most challenging animations to create as it required multiple distinct state changes, each with its own set of animations. Three objects needed to be animated to achieve this effect: the arms, the pan, and the meat.

```
case 0:  
    rotcheff += 0.3f;  
    poszsar += 0.1;  
    rotsarten += 0.25;  
    carnez = -15.0 * cos(carneinc);  
    carney = 12.0 * sin(carneinc);  
    carneinc += 0.015;  
    if (carneinc >= 3) { //3  
        carneinc = 3.0f;    //1.5  
    }  
    if (rotcheff >= 30)  
        animcheff = 1;  
    break;  
case 1:  
    carneinc += 0.04;  
    if (carneinc >= 3.1) {  
        animcheff = 2;  
    }  
    break;
```

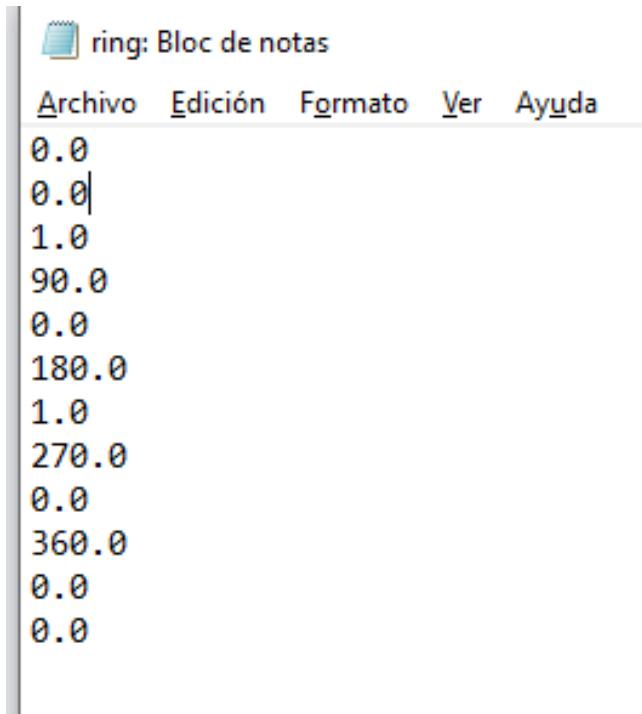
```
case 2:  
    carnez = -15.0 * cos(carneinc);  
    carney = 70 * sin(carneinc);  
    carneinc += 0.005;  
    rotcheff -= 0.3f;  
    rotsarten -= 0.25;  
    poszsar -= 0.1;  
  
    if (rotcheff <= 0) {  
        animcheff = 3;  
    }  
    break;  
case 3:  
    carnez = -15.0 * cos(carneinc);  
    carney = 70 * sin(carneinc);  
    carneinc += 0.008;  
    if (carneinc >= 4.85) {  
        animcheff = 4;  
    }  
    break;  
case 4:  
    carneinc += 0.01;  
    if (carneinc >= 7) {  
        carneinc = 1.5f;  
        animcheff = 0;  
    }  
    break;
```

The animations were synchronized to create a seamless throwing motion. In the first case, movement was applied to the arm, pan, and meat simultaneously. In the second case, only the meat was animated. In the third case, the meat continued its movement while the pan and chef returned to their original positions. Finally, in the fourth state, the meat reached its destination, followed by a fifth state delay before restarting the animation.

ANIMATIONS

- "RING KEYFRAMES"

For the rings, a simple animation was created using keyframes. The animation data was obtained from an external file called "rings.txt," which contains the values for the animation.



The screenshot shows a Windows-style notepad window titled "ring: Bloc de notas". The menu bar includes "Archivo", "Edición", "Formato", "Ver", and "Ayuda". The main content area contains the following text:

```
0.0
0.0
1.0
90.0
0.0
180.0
1.0
270.0
0.0
360.0
0.0
0.0
```

To obtain the coordinates from the file and use them for keyframes, the following code was used:

```
string arch = "ring.txt";
ifstream archivo(arch.c_str());
string lineas;

while (getline(archivo, lineas)) {

    printf("%i ", i);
    switch (i) {
    case 0:
        KeyFrame[pos].poszring = std::stof(lineas);
        break;
    case 1:
        KeyFrame[pos].rotring = std::stof(lineas);
        break;

    }
    if (i < 2) {
        i++;
    }
    if (i == 2) {
        pos++;
        i = 0;
    }

}
```

ANIMATIONS

For this animation, there are a total of 5 keyframes, so the maximum value was set to 4.

```
FRAME KeyFrame[MAX_FRAMES];
int FrameIndex = 4;
bool play = false;
int playIndex = 0;
```

A structure called _frame was used for the keyframes, which consists of 2 data points.

```
typedef struct _frame
{
    //Variables para GUARDAR Key Frames
    float poszring;
    float rotring;

}FRAME; //Estructura Frame
```

For this program, a variable was needed for the rotation of the ring and another one for its up and down movement. Additionally, two more variables were necessary for the increments.

```
//Rings Keyframes
float rotring = 0.0f;
float poszring = 0.0f;
//Calculo incremento
float incrotring = 0.0f,
incposz = 0.0f;
```

For this animation, the goal was to achieve a similar look to what is seen in games, which is why the movements are kept simple.

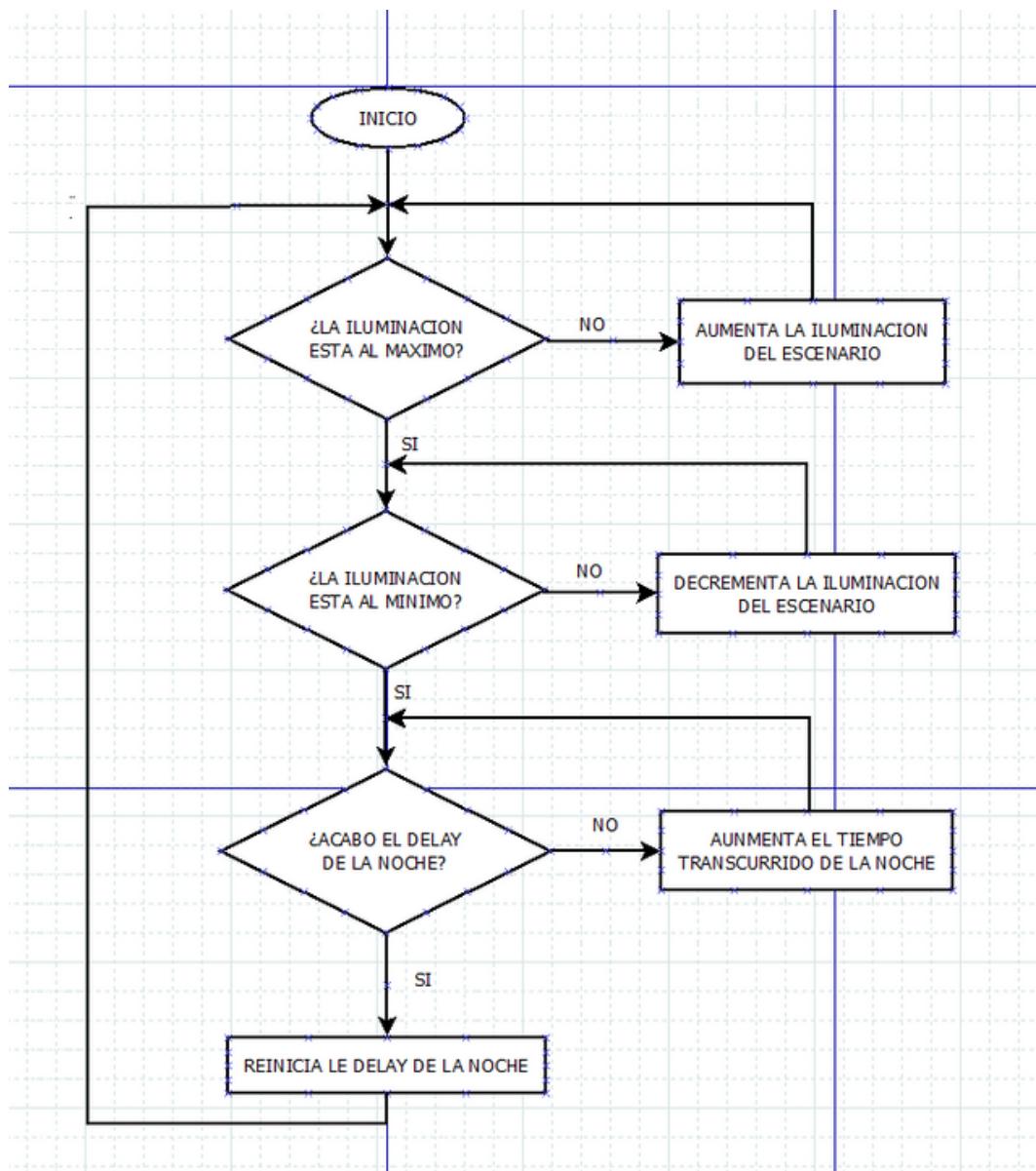
ANIMATIONS

To make more realistic

ILUMINATION

- "CICLE DAY AND NIGHT"

To create the day and night cycle, we aimed for an approximate duration of one minute. To develop the state machine that would allow us to achieve this animation, we used the following flowchart.



For the day and night cycle, we used ambient lighting and aimed for a white light. The maximum illumination value was set to 0.8, while the minimum value was set to 0.1.

The duration of the day and night cycle is determined by the increase and decrease in light values. In this case, we used an increment and decrement value of 0.0005.

ILUMINATION

- "Automatic Spotlight On AND Off"

Automatic Light On and Off synchronized with Day and Night Cycle

To achieve automatic on and off functionality for the light, it was synchronized with the day and night cycle using the same flowchart.

During the daytime, the light is programmed to have a diffuse light value of 0.55, aiming to create a yellowish hue, and a specular light value of 0.2, providing a white light effect.

During the nighttime, the light transitions to a diffuse and specular light value of 0, resulting in only ambient light. The light being off signifies that the establishment is closed, hence there is no illumination.

- "SPOTLIGHT ON AND OFF VIA KEYBOARD"

Toggle On and Off Light controlled by a Button positioned on the arcade screen. It was set to emit an intense red light with a limited spread.

The red light's range was reduced to achieve a close range effect, rather than a widespread illumination like that of a lamp. Additionally, the intention was to give it a sinister appearance if it is turned on during the night.

BIBLIOGRAFIA

mosassauro. (Apr 19, 2016). *TurboSquid*. Recuperado de <https://www.turbosquid.com/es/3d-models/blender-fbx-free/1028759>

HuNtEr_3DdD. (Feb 15, 2023). *TurboSquid*. Recuperado de <https://www.turbosquid.com/es/3d-models/blender-fbx-free/2028528>

Monroe Henry. (Sep 1, 2022). *TurboSquid*. Recuperado de <https://www.turbosquid.com/es/3d-models/blender-fbx-free/1952538>

s.a. (s.f). *The Models Resource*. Recuperado de https://www.models-resource.com/ds_dsi/sonicrush/model/11252/

s.a. (s.f). *The Models Resource*. Recuperado de https://www.models-resource.com/ds_dsi/sonicrush/model/1028759/

s.a. (s.f). *The Models Resource*. Recuperado de https://www.models-resource.com/ds_dsi/sonicrush/model/32752/

s.a. (s.f). *The Models Resource*. Recuperado de https://www.models-resource.com/ds_dsi/sonicrush/model/38513/

BIBLIOGRAFIA

s.a. (s.f). *The Models Resource*. Recuperado de
[https://www.models-
resource.com/ds_dsi/sonicrush/model/58672/](https://www.models-resource.com/ds_dsi/sonicrush/model/58672/)

s.a. (s.f). *The Models Resource*. Recuperado de
[https://www.models-
resource.com/ds_dsi/sonicrush/model/32754/](https://www.models-resource.com/ds_dsi/sonicrush/model/32754/)

s.a. (s.f). *The Models Resource*. Recuperado de
[https://www.models-
resource.com/ds_dsi/sonicrush/model/14681/](https://www.models-resource.com/ds_dsi/sonicrush/model/14681/)

s.a. (s.f). *The Models Resource*. Recuperado de
[https://www.models-
resource.com/ds_dsi/sonicrush/model/47629/](https://www.models-resource.com/ds_dsi/sonicrush/model/47629/)

s.a. (s.f). *The Models Resource*. Recuperado de
[https://www.models-
resource.com/ds_dsi/sonicrush/model/36850/](https://www.models-resource.com/ds_dsi/sonicrush/model/36850/)

s.a.(s.f).cgtrader.
<https://www.cgtrader.com/items/3730295/download-page>

Sketchfab. (n.d.). Bonnie The Bunny - Download Free 3D model by William Afton (Co Owner of Fredbears Family Diner) (@davismav). <https://sketchfab.com/3d-models/bonnie-the-bunny-1c7953d6d5bb4f458164cb87965c596b>

Audio

The Living Tombstone (31 ago 2014). Five Nights at Freddy's 1 canción - The Living Tombstone cancion original. You Tube.

<https://www.youtube.com/watch?v=l18A5BOTIzE>