



SCHOOL OF ENGINEERING, UNAM



FINAL PROJECT

TECHNICAL MANUAL

Computer Graphics and
Computer-Human Interaction

Student ID:

317170115

Group:

13

Course:

2024-2

CONTENT

1.	Objectives.....	2
2.	Project Scope.....	3
3.	Limitations.....	3
4.	Costs.....	5
5.	Time	6
6.	Software flowchart	7
7.	Software flowchart	12
8.	Code Documentation	
8.1.	Libraries and Headers	15
8.2.	Functions.....	15
8.3.	Variables	16
8.4.	Shaders	16
9.	Conclusions.....	19

OBJECTIVES

General

- The student must apply and demonstrate the knowledge acquired throughout the course, with the following considerations:
 - Select a facade and a space (real or fictional).
 - Present reference images of these spaces.
 - The selected spaces must be recreated in a 3D environment in OpenGL.
 - The reference images must include 7 objects suitable for the environment, adding realism and dynamism.
 - The 7 selected elements must be recreated within an ambient space, which will also be inside the corresponding facade.
 - Aim for good modeling and apply good techniques of texturing, animation, and lighting to achieve greater realism according to the presented image and what is sought to be recreated.
 - Include 5 animations (3 simple, 2 complex) within the environment being worked on.

Particular

- Taking the general objective as a starting point, the student will create a 3D environment inspired by a toy store of the American enterprise **Toys 'R' Us**, where characteristic objects of toy stores will also be recreated, such as shelves with toy boxes, shelves with stuffed animals, large-scale figures to decorate the interior, etc. Some of the mentioned objects will be animated, having on count the general project requirements.

With the above considerations, the student will use and apply different forms of computer graphics creation and rendering, reinforcing the knowledge obtained throughout the course adapted to a personalized form.

PROJECT SCOPE

The present project has a vast academic scope that allows us to apply and reinforce all the knowledge acquired throughout the semester, in addition to the flexibility with which it's approached since it begins to be worked on several weeks before its delivery. I think, it's a very good option if you're interested in model, texture, and adapt models representing anything from a real or fictional environment, having the freedom to do so in the way we desire and is most comfortable for each individual student, because there are so many ways and techniques to accomplish most of the requested tasks, allowing it to be a highly customizable experience. It works as a way for us, as students, to understand and have solid foundations in the necessary knowledge to carry out creations with a higher degree of complexity. Adapting our work to a set of requirements and making them acquire a certain complexity, undoubtedly, forges us in a very solid way for our future as professionals. Also I have to say that, by understanding the functioning and use of OpenGL, we can use it to carry out projects that involve a necessarily user-friendly and attractive graphical interface for users in general, so working with OpenGL in this project will also give us skills for creating software with user interfaces.

There are no doubts about that this project can greatly help us lay the necessary things to be able to create and work on larger projects since this project gives us the necessary abstraction ability to be able to recreate scenarios and objects within a digital space, and this abstraction ability will help us develop a more logical and abstract thinking, useful in the performance of any task that involves problem-solving through the development of a creative mind.

LIMITANTES

This project has certain limitations caused by various factors that must be considered for its development and subsequent use, below are listed and punctually described some of these limitations:

- **Computer Equipment Limitations:** Some projects will require extensive processing power in terms of the objects that make up the project itself, a lot of the students enrolled in the faculty doesn't have the possibility to purchase strong computers; but talking about it personally, my computer doesn't have the capability to render highly detailed objects or complex lighting techniques in a project of this level. Actually, most of the limitations described below stem from this limitation.
- **Graphics Realism:** As students who had a first approach to modeling in this course, we have the limitation of our current skills to model objects with a very high level of realism and polygon count, this is also compounded by the fact that some of us don't have high-powered equipment to process highly detailed objects. In my case, I had to modify some of my objects because they were not loading correctly into OpenGL, causing the project to take too long to compile and ultimately Visual Studio would automatically close it. For reasons like the ones mentioned, we can't generate

incredibly realistic graphics, modeling with a high level of detail implies a higher polygon count, which makes the level and time of processing or rendering quite large, making it difficult for our computers to do this in a considerably short time.

- **Lightning and Animations:** Lighting techniques are also a very heavy variable in computer graphics processing, the more realistic the lighting, the heavier the environment is to load for some computer equipment. However, the present project aims to achieve good and realistic lighting without being too heavy on loading. In terms of animations, some are complex considering what they imply for their correct functioning in contrast to what we worked on throughout the course; with this in mind, animations were chosen that fit well with the context of the environment and thus could add more dynamism to the project.

COSTS

Here's a simulation of the costs involved in the development of the project for a more realistic environment. These are described by concept to calculate the total sum in the following diagram:

ID	CONCEPTO	COSTO
PRP-1	Proposal - reference images	\$125.00
MDL-1	Modeling and texturing of structure	\$250.00
MDL-2	Modeling and texturing of object 1 (Master Chief Statue)	\$75.00
MDL-3	Modeling and texturing of object 2 (Darth Vader Statue)	\$75.00
MDL-4	Modeling and texturing of object 3 (Toy Train)	\$75.00
MDL-5	Modeling and texturing of object 4 (Checkout Area)	\$75.00
MDL-6	Modeling and texturing of object 5 (Plush Pokémon)	\$75.00
MDL-7	Modeling and texturing of object 6 (Toy Shelves)	\$75.00
MDL-8	Modeling and texturing of object 7 (Basketball Hoop)	\$75.00
EXP-1	Loading models and environment in OpenGL	\$225.00
ILUM-1	Lighting in OpenGL	\$100.00
ANIM-1	Animation programming	\$250.00
MNL-1	Technical manual (Spanish and English)	\$125.00
MNL-2	User manual (Spanish and English)	\$125.00
SFTW-1	Modeling software license (AutoDesk Maya 2023)	<i>included (student license)</i>
SFTW-2	Programming software license (Microsoft Visual Studio)	<i>included (community version)</i>
SFTW-3	Texturing software (GIMP)	<i>included (Open Source)</i>
TOTAL		\$1,725.00 USD

TIME

The project finalization time is graphically represented by the following Gantt chart, with an approximate duration of 8 weeks for its development until the agreed delivery date, which is scheduled for May 10th, 2024:

	WEEK (DD-MM)							
	1	2	3	4	5	6	7	8
ID	15/03 - 22/03	22/03 - 29/03	29/03 - 05/04	05/04 - 12/04	12/04 - 19/04	19/04 - 26/04	26/04 - 03/05	03/05 - 10/05
PRP-1								
MDL-1								
MDL-2								
MDL-3								
MDL-4								
MDL-5								
MDL-6								
MDL-7								
MDL-8								
EXP-1								
ILUM-1								
ANIM-1								
MNL-1								
MNL-2								
DELIVERY								may 10th

OBJECTS MODELING AND TEXTURING

This section presents the decisions made for the modeling of the store and the objects that will be present in the 3D environment. In case they were acquired from another source, the corresponding credits are provided.

- **Toys 'R' Us Store**

The store was fully modeled and textured by the student using Autodesk Maya software.



- **Basketball Hoop**

This object was recreated using artificial intelligence ([Luma AI \(lumalabs.ai\)](https://lumalabs.ai)), with the following prompt: “*basketball hoop for kids, 2 meters tall with its own base*”



- **Master Chief Statue (Halo character)**

"PC Computer - Fortnite - Master Chief" (<https://skfb.ly/6ZynF>) by HarrisonHag1 is licensed under Creative Commons Attribution (<http://creativecommons.org/licenses/by/4.0/>).

The model of Cortana in Master Chief's hand was added using Autodesk Maya and: "Characters>Halo 4>Cortana" (<https://skfb.ly/oNRWK>) by jameslucino117 is licensed under Creative Commons Attribution-NonCommercial (<http://creativecommons.org/licenses/by-nc/4.0/>).



- **Toy Shelves**

These objects, and the toy boxes they contain, were recreated and textured by the student using Autodesk Maya software.

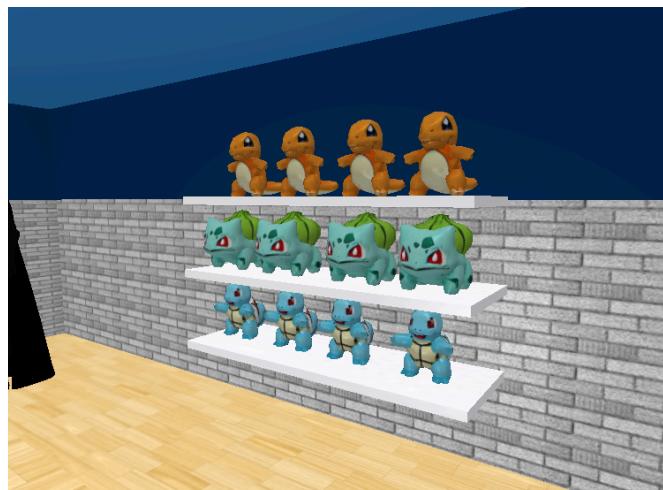


- **Plush Pokémon on Shelves**

The different Pokémon were recreated using artificial intelligence ([Luma AI \(lumalabs.ai\)](https://lumalabs.ai)), with the following prompts:

- "charmander plush"
- "bulbasaur plush"
- "squirtle plush"

Las repisas que los sostienen fueron creadas y texturizadas por el alumno con el software Autodesk Maya.



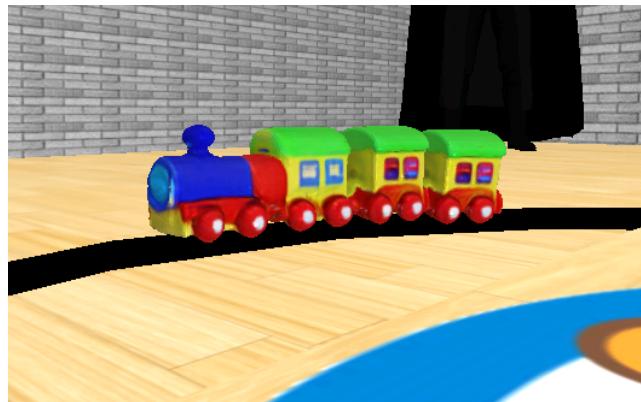
- **Darth Vader Statue (Star Wars character)**

"Darth Vader" (<https://skfb.ly/onpSP>) by HarrisonHag1 is licensed under Creative Commons Attribution (<http://creativecommons.org/licenses/by/4.0/>).



- **Toy Train**

This object was recreated using artificial intelligence ([Luma AI \(lumalabs.ai\)](#)), with the following prompt: “



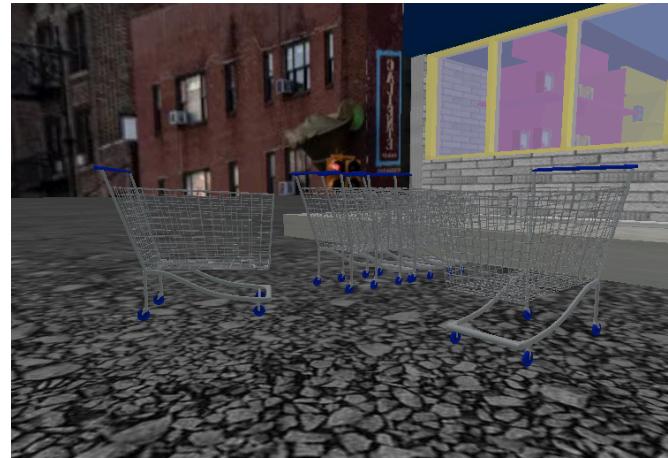
- **Checkout Area**

This object, and the toy boxes it contains, were recreated and textured by the student using Autodesk Maya software.



- Shopping Cart

"3d Supermarket Cart Model" ([3d Supermarket Cart Model \(turbosquid.com\)](#)) by [KaianFS](#) is licensed under TurboSquid ([The TurboSquid 3D Model License - TurboSquid Blog](#)).



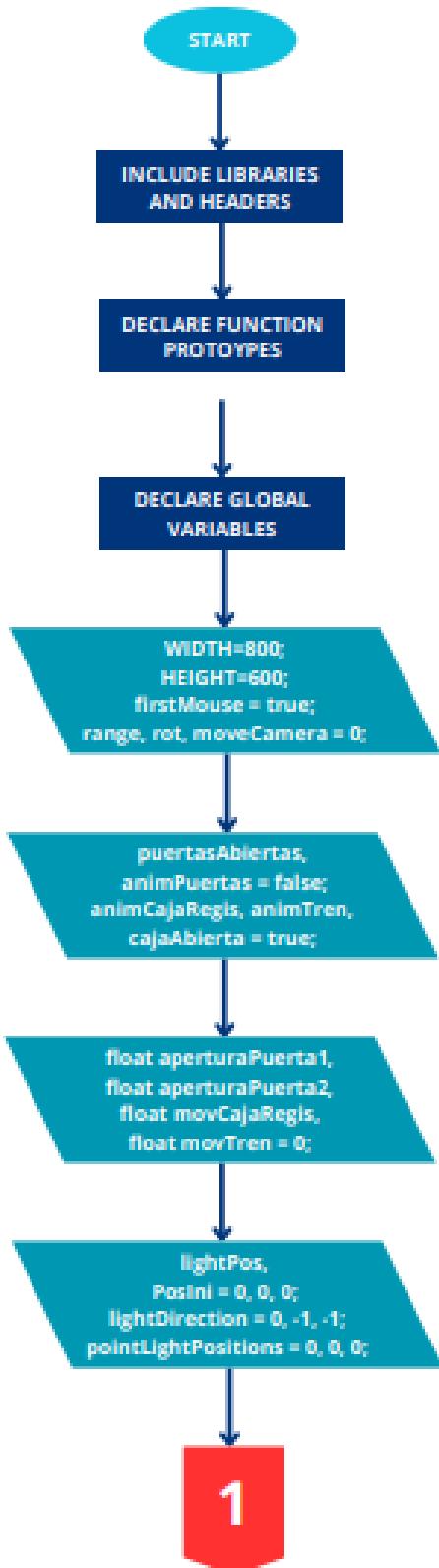
- Dove

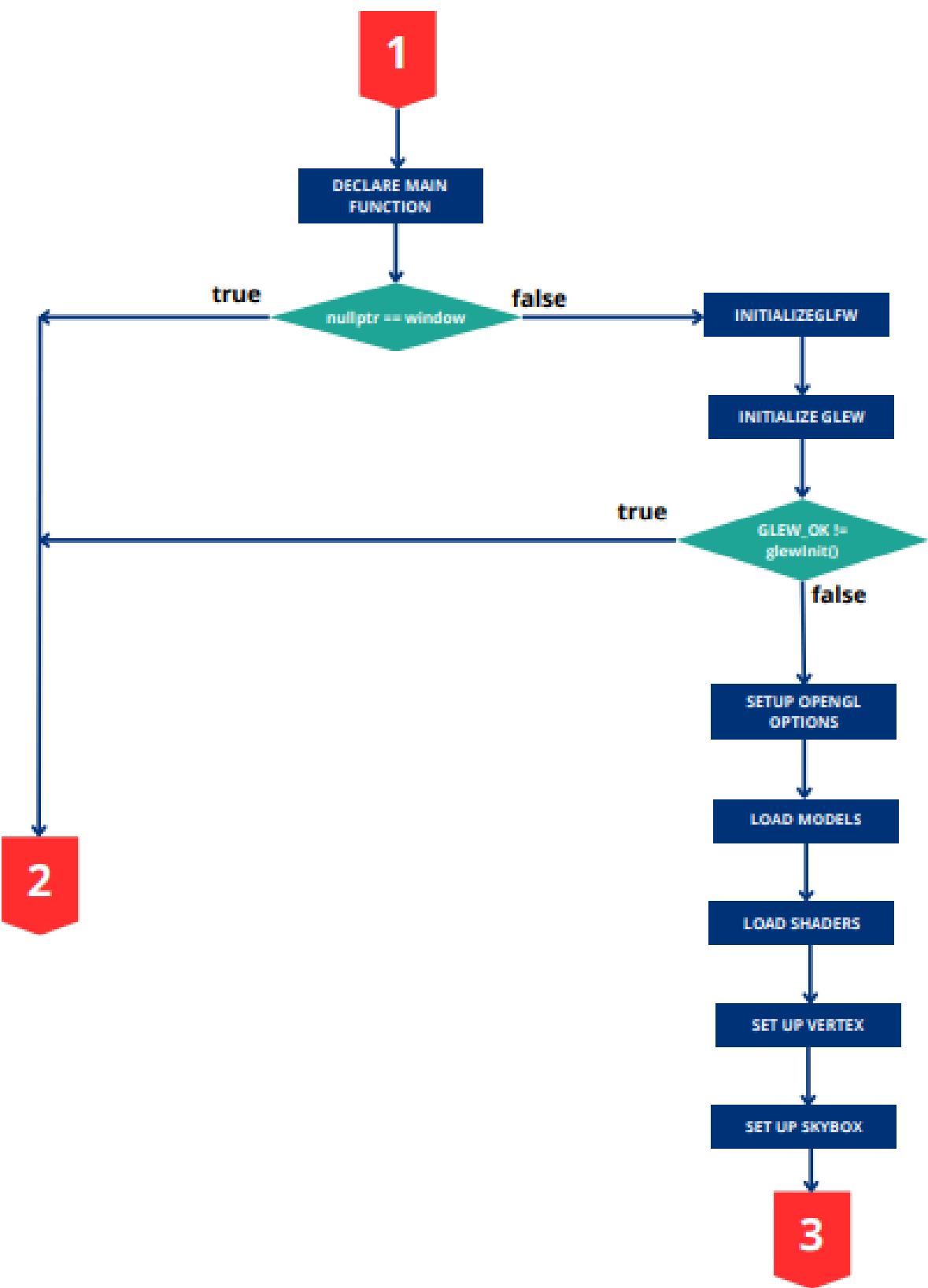
"Flying Dove Bird" ([hModelo 3d sin pájaro de la paloma voladora - .Obj - Open3dModel](#)) by N/A is licensed under Open 3D Model ([Política de privacidad | Open3dModel.com](#)).

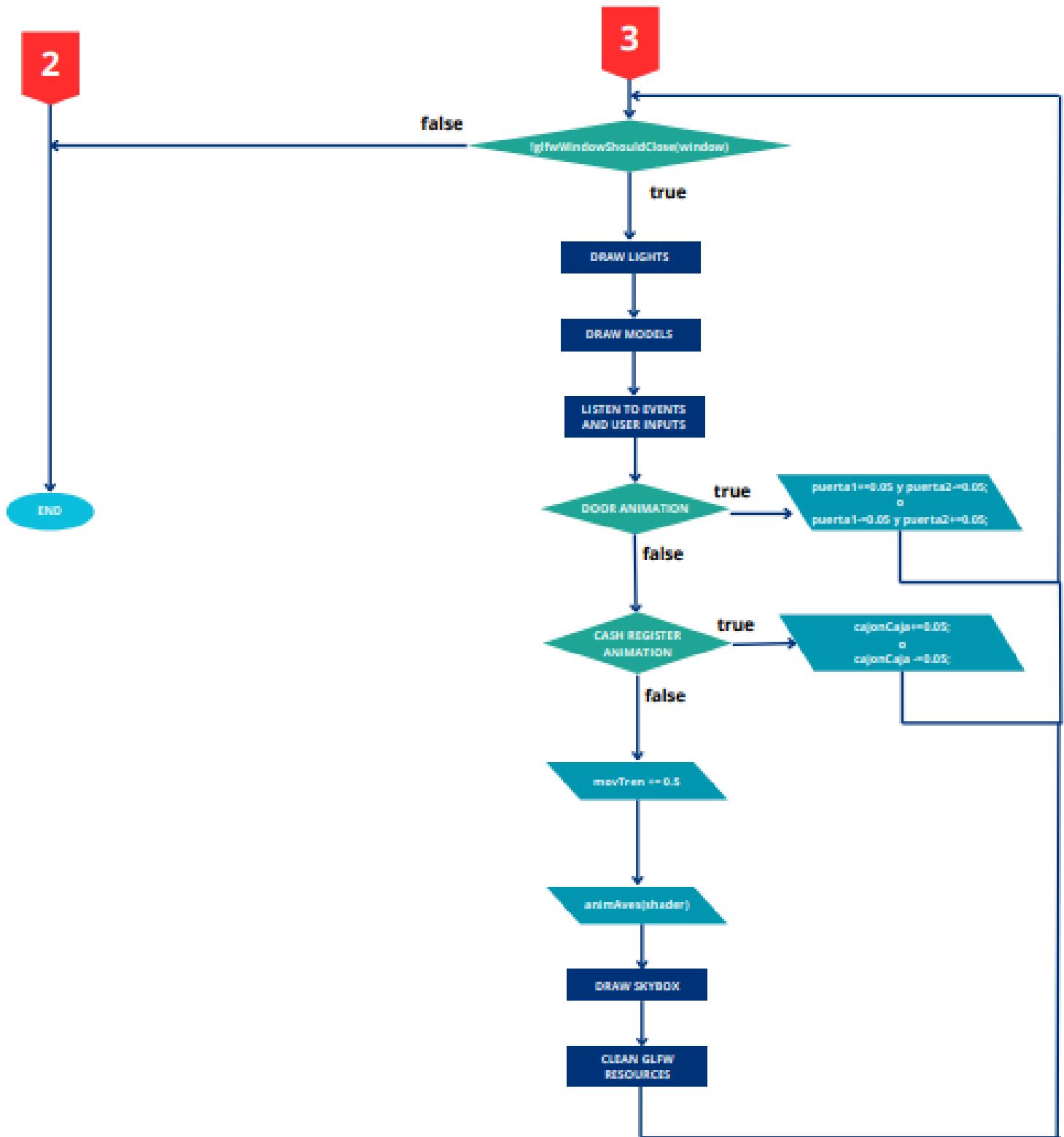


SOFTWARE FLOWCHART

Below is the flowchart representing the program in a general way, this helps to provide a better understanding of what happens during runtime:







CODE DOCUMENTATION

This entire section was done in and with the help of Microsoft Visual Studio 2023 Community Version software. It mentions and explains the different libraries, variables, and functions used, necessary for the overall operation of the program:

Libraries

TYPE	NAME	USAGE
Library	iostream	Processing of input and output in the form of a sequence of bytes. Input from a device to main memory and output to the screen.
Library	cmath	Set of functions to facilitate the implementation of mathematical operations.

Headers

TYPE	PATH/NAME	USAGE
Header	GL/glew.h	OpenGL extensions compatible with the target platform.
Header	GLFW/glfw3.h	Window creation and manipulation.
Header	stb_image.h	Image processing.
Header	glm/glm.hpp	Mathematical functions within OpenGL.
Header	glm/gtc/matrix_transform.hpp	Functions for matrix transformation in a simpler and more efficient way.
Header	glm/gtc/type_ptr.hpp	Interaction between vectors, matrices, and pointers.
Header	SOIL2/SOIL2.h	Texture loading in OpenGL.
Header	Shader.h	Program designed to run in graphics processing.
Header	Camera.h	Virtual position of the user

		within the 3D environment.
Header	Model.h	Models loading in OpenGL.
Header	Texture.h	Texture loading in OpenGL.

Functions

TIPO	NOMBRE	USO
void	KeyCallback()	Reads keyboard inputs.
void	MouseCallback()	Reads mouse-movement inputs.
void	DoMovement()	Camera movement as an user input.
void	animacion()	Use of the “animación” animation.
int	main()	Main flow of the program, where models, shaders, lighting, animations, etc., are loaded.

Variables

PROPÓSITO	TIPO	NOMBRE	USO
Ventana	const GLuint	WIDTH	Window width.
Ventana	const GLuint	HEIGHT	Window height.
Ventana	int	SCREEN_WIDTH	Assign window width.
Ventana	int	SCREEN_HEIGHT	Assign window height.
Cámara sintética	Camera	camera	Camera movement as an user input.
Cámara sintética	GLfloat	lastX	Camera position on the X-axis.
Cámara sintética	GLfloat	lastY	Camera position on the Y-axis.
Cámara sintética	bool	keys	Key-press recognition.
Cámara sintética	bool	firstMouse	Mouse control for

			keeping it on the center of the screen.
Cámara sintética	float	range	Camera's FOV range
Cámara sintética	float	rot	Camera's rotation value.
Cámara sintética	float	movCamera	Values for moving and transforming camera.
Animaciones	bool	puertasAbiertas	If false, it opens the doors when the button is pressed; if true, closes the doors when the button is pressed.
Animaciones	bool	animPuertas	Recognizes when the button is pressed to activate the door animation.
Animaciones	bool	animTren	Recognizes when the button is pressed to activate the door animation.
Animaciones	bool	cajaAbierta	If false, it opens the cash register drawer when the button is pressed; if true, closes the cash register drawer when the button is pressed.
Animaciones	bool	animCajaRegis	Recognizes when the button is pressed to activate the cash register animation.
Animaciones	float	aperturaPuerta1	Defines the opening range of the right door.
Animaciones	float	aperturaPuerta2	Defines the opening range of the left door.
Animaciones	float	movCajaRegis	Defines the opening range of the cash register drawer.
Animaciones	float	movTren	Defines the speed of the train movement.
Animaciones	float	tiempo	Helps the AnimAves shader for the movement of flying birds.
Animaciones	float	speed	Helps the AnimAves shader for the speed of flying birds.

Animaciones	float	movCarroCompras	Determines the movement of the shopping cart when the key is pressed.
Iluminación	glm::vec3	lightPos	Light position.
Iluminación	glm::vec3	PosIni	Initial light position.
Iluminación	glm::vec3	lightDirection	Light direction.
Iluminación	glm::vec3	pointLightPositions[]	Array containing positions of different point lights.

Shaders

NOMBRE	USO
lightingShader	Used to draw models (solid or transparent, depending on the alpha channel) with defined lighting.
lampShader	Support for defined lighting.
SkyBoxShader	Used to draw the environment's skybox, which is like an aesthetic background.
animShader	Solid object animation.
AnimAves	Keeps birds flying over the toy store while the project is running.

CONCLUSIONS

As mentioned in the scope section, this project equips us with skills and reinforces them for the subject in general. The applied concepts and the way they are implemented are of great use for the student to develop their logic, skills, and abstraction ability.

In general, there were no major complications since almost 100% of the things that make up this project were seen, done, and practiced during the course. The exceptions are minimal and focus more on a personalized experience for the student given the creative freedom granted to us in doing this work. Speaking personally, I consider some concepts such as lighting and complex animations to still be somewhat confusing; however, these concepts were still successfully implemented to meet the objectives set from the beginning.

In conclusion, I must mention that I am very satisfied with what I have learned so far from the program and, of course, from the teacher. Without a doubt, his method of explanation and practice, in addition to his support at all times, were fundamental aspects in achieving success in the objectives. Although it was lengthy, it was a very enjoyable and interesting project.