Technical Manual

HOUSE Phineas and Ferb

INDICE

Project Objective

The purpose of this project is to apply the knowledge acquired during the Computer Graphics course by creating a three-dimensional environment inspired by the house from the *Phineas and Ferb* series, which includes two rooms: a living room and a kitchen.

To achieve this, geometric modeling techniques will be used with basic primitives, applying textures through graphic libraries. The objective is to generate a visually accurate representation, with attention to detail and faithfulness to the original design.

Additionally, animation principles and graphical ambiance will be implemented to simulate a dynamic and interactive space. Both simple and more complex animations will be developed to reflect real-world conditions, such as the transition from day to night or smoke rising from the chimney. These animations will either react to the environment or run automatically to enhance realism.

To enrich the visual experience, advanced lighting models will be integrated to add depth and definition, along with effects such as transparency and shadows. These features will contribute to a more immersive and believable atmosphere for the user.

In summary, this project aims to bring together techniques, tools, and concepts learned throughout the course to build a visually attractive environment, based on a cartoon, allowing the user to interact with a faithful and detailed representation of the house.

Project Scope

The project is based on the use of the OpenGL graphics API, which allows handling of 3D graphics and the application of realistic visual effects. OpenGL will be combined with advanced rendering techniques to deliver a smooth and high-quality experience.

The language used will be C++, which provides more efficient control over memory and low-level graphic processes. This will optimize the program's performance, especially in managing visual resources.

Development will be carried out using Visual Studio Community 2022, leveraging its debugging, optimization, and version control tools to maintain an organized and stable workflow.

Among the animated elements included from the program's start are the chimney smoke and day-night lighting cycles. These animations will run automatically, contributing to a continuous and realistic ambiance from the beginning of the experience.

Project Costs

1. Direct Costs

1.1 Specialized Labor

- **Facade Design**
  - Estimated Time: 36 hours (9 weeks, 4 hours/week)
  - Hourly Rate: $180 MXN
  - Subtotal: $6,480 MXN
- **Object Modeling**
  - Total Objects: 12
  - Estimated Time: 48 hours (4 hours/object)
  - Hourly Rate: $200 MXN
  - Subtotal: $9,600 MXN
- **Texturing and OpenGL Testing**
  - Estimated Time: 24 hours
  - Hourly Rate: $190 MXN
  - Subtotal: $4,560 MXN
- **Lighting Configuration**
  - Spotlight (Directional Light Source):
    - Requires specific technical knowledge
    - Subtotal: $1,200 MXN
  - Point Lights:
    - Subtotal: $1,000 MXN
- **Animations**
  - Complex Animations (2 units):
    - Estimated Time: 16 hours (8 hours each)
    - Hourly Rate: $200 MXN
    - Subtotal: $3,200 MXN
  - Simple Animations (3 units):
    - Estimated Time: 9 hours (3 hours each)
    - Hourly Rate: $180 MXN
    - Subtotal: $1,620 MXN

1.2 Technical Resources and Software

- **Hardware**: A high-performance computer was used throughout the process.
- **Electricity**: Estimated cost based on usage hours.

2. Indirect Costs

2.1 Assets

- **Software Used**:
  - Autodesk 3Ds Max (educational license): Free
  - OpenGL: Free

2.2 Liabilities

- **Associated Expenses**:
    - Electricity

3. Value-Added Tax (VAT)

- VAT calculated on direct cost subtotal:
    - Subtotal: $27,660 MXN
    - VAT Rate: 16%
    - VAT: $4,425.60 MXN

4. Total Project Cost

- Subtotal Direct Costs: $27,660 MXN
- VAT (16%): $4,425.60 MXN
- Total: $32,085.60 MXN

Cost Summary

CONCEPT	SUBTOTAL (MXN)	VAT (16%)	TOTAL (MXN)
Facade Design	$6,480	$1,036.80	$7,516.80
Object Modeling	$9,600	$1,536.00	$11,136.00
Texturing and OpenGL Tests	$4,560	$729.60	$5,289.60
Lighting Configuration	$2,200	$352.00	$2,552.00
Animations	$4,820	$771.20	$5,591.20
TOTALS	**$27,660**	**$4,425.60**	**$32,085.60**

Constraints

These constraints define the necessary limitations for the program’s correct functionality and performance:

- **Maximum Object Size**: 3D objects used in the scene have size limitations to ensure proper performance and avoid overloading GPU memory.
- **Textures Scaled to Powers of Two**: Textures must be scaled to power-of-two dimensions to ensure proper reading and processing in OpenGL, avoiding rendering issues and ensuring compatibility with API specifications.

Table of Variables and Descriptions

Variable	Type / Constant	Description
toRadians	const float	Constant to convert degrees to radians.
solDirZ	float	Sun direction on the Z-axis (for day/night cycle).
solDirY	float	Sun direction on the Y-axis.
solDirYOffset	float	Additional Y-axis offset for the sun.
invierteCiclo	bool	Indicates if the day/night cycle is being reversed.
esDeDia	bool	Flag indicating if it's currently daytime.
humoTime	float	Accumulated time for smoke animation.
humoDuration	float	Total smoke cycle duration.
humoScale	float	Initial smoke scale.
humoMaxScale	float	Maximum smoke scale.
humoHeight	float	Initial smoke height.
humoMaxHeight	float	Maximum smoke height during animation.
humoOpacity	float	Smoke opacity level.
puertaAbierta	bool	Indicates if the door is open.
anguloPuerta	float	Current rotation angle of the door.
velocidadPuerta	const float	Speed at which the door opens/closes.
animDurationDados	float	Dice animation duration.
velocidadRotacion	float	Dice rotation speed.
alturaMaxima	float	Maximum height dice reach when “jumping.”
posDado1	const glm::vec3	Fixed position of dice 1 on the table.
posDado2	const glm::vec3	Fixed position of dice 2 on the table.
rotFinalDado1	static glm::vec3	Final rotation of dice 1.
rotFinalDado2	static glm::vec3	Final rotation of dice 2.
animTimeDados	float	Accumulated time for dice animation.
alturaDados	float	Current height of dice during animation.
mainWindow	Window	Main rendering window.
meshList	std::vector<Mesh*>	List of loaded meshes.
shaderList	std::vector<Shader>	List of compiled shaders.
camera	Camera	Cameras used in the scene.

Variable	Type / Constant	Description
pisoTexture	Texture	Texture used for the floor.
HumoTexture	Texture	Texture used for the smoke.
Casa_Phineas	Model	3D model of Phineas' house.
Puerta	Model	3D model of the door.
Dado1, Dado2	Model	3D models of the dice.
skybox, skybox2	Skybox	Skyboxes (one for day, one for night).
Material_brillante	Material	Shiny material (high specularity).
Material_opaco	Material	Dull material (low specularity).
deltaTime	GLfloat	Time between frames.
lastTime	GLfloat	Time of the last frame.
limitFPS	static double	FPS limit (60).
mainLight	DirectionalLight	Main directional light source.
pointLights	PointLight[MAX_POINT_LIGHTS]	Array of point lights.
spotLights	SpotLight[MAX_SPOT_LIGHTS]	Array of spotlight-type lights.
vShader	const char*	Vertex shader path.
fShader	const char*	Fragment shader path.

Recreated Scenarios

Kitchen

Applied Software Methodology

The development of this project followed the **Scrum agile methodology**, one of the most widely used in software engineering due to its iterative and incremental approach. This methodology allowed us to organize the work into **sprints**, which are short, defined development cycles with specific and measurable goals.

At the end of each sprint, progress was reviewed and tasks and priorities were adjusted based on the progress made and any challenges encountered. This dynamic enabled continuous improvement of design, programming, and implementation, while also fostering constant communication within the team.

Thanks to Scrum, it was possible to detect errors early, make flexible adjustments, and ensure a functional delivery aligned with project requirements.

Conclusion

This project was a valuable opportunity to strengthen and apply the knowledge acquired in Computer Graphics. Throughout the development process, I not only applied tools learned during the course but also integrated previous knowledge such as structured planning and diagramming to organize the workflow.

Facing the challenge of building a realistic 3D environment pushed me to conduct independent research and incorporate additional techniques, such as creating custom variables to manage complex animations. As a result, I was able to represent visual effects like fire and smoke with careful attention to movement for a convincing appearance.