GSoC Proposal: Rendering and Graphics

Project Title: Enhancing Real-Time Rendering and UI/UX Design for Open-Source Rendering Engines

Name:Anshu Kumari

Email:kumarianshu301130@gmail.com

GitHub/Portfolio:anshu-codes

University:Institute of Management Studies,Noida

Degree Program:Bachleor of Computer Applications

Year of Study:2nd year

Abstract

This proposal aims to contribute to the development of an open-source rendering engine by implementing advanced rendering techniques while also improving the user interface (UI) and user experience (UX) design for asset management and visualization. With a solid foundation in computer graphics, game development, and UI/UX design tools such as Canva, Figma, Adobe Creative Cloud (Illustrator, Photoshop, InDesign), and Inkscape, this project will focus on creating a seamless user experience for managing graphical assets, optimizing the rendering pipeline for better performance, and introducing visually appealing rendering effects. The goal is to strike a balance between high-quality visual output

and an intuitive interface for both developers and non-technical users.

Benefits to the Community

The combination of optimized rendering techniques and a robust UI/UX design will significantly enhance the usability and visual quality of open-source rendering engines. By improving asset management, rendering performance, and visual effects, this project will allow artists, developers, and other users to collaborate more effectively. Additionally, the project's focus on clean UI design and asset visualization will create a more engaging and user-friendly environment, ensuring that graphical and performance improvements are easily accessible to a broader audience, including non-developers.

Deliverables

1. Enhanced Rendering Pipeline: Implement and optimize rendering algorithms such as physically-based rendering (PBR), ray tracing, and post-processing effects like bloom, ambient occlusion, and motion blur.

2. UI/UX Design for Asset Management: Design and integrate a user-friendly interface for managing textures, materials, and other graphical assets, using Figma, Canva, and Adobe tools for prototyping and UI design.

3. Shader Implementation: Develop and integrate PBR shaders and other advanced rendering techniques to improve the realism of 3D scenes.

4. Optimized Performance: Focus on improving GPU utilization and overall rendering performance to ensure real-time frame rates, even with complex scenes and effects.

5. Comprehensive Documentation and Tutorials: Provide clear, concise documentation and tutorials for developers and artists to utilize the rendering engine and its new UI/UX features effectively.

Technical Approach

1. Initial Setup and Research:

- Familiarize myself with the existing codebase of the open-source rendering engine.

- Study the rendering pipeline to identify potential areas for optimization and enhancement.

- Research modern techniques in graphics rendering, such as PBR, real-time ray tracing, and advanced post-processing effects.

2. Rendering Pipeline Enhancements:

- PBR Shaders: Implement realistic material rendering using PBR, allowing for more accurate surface interactions, including diffuse, specular, and reflective properties.

- Post-Processing Effects: Develop effects like bloom, ambient occlusion, motion blur, and depth of field to add realism and visual interest to the rendered scenes.

- Real-Time Ray Tracing: Explore integrating ray tracing techniques to improve lighting, shadows, and reflections, focusing on performance optimizations to run in real-time.

3. UI/UX Design and Integration:

- Asset Management Interface: Create a clean, intuitive UI for managing graphical assets such as textures, shaders, and 3D models. This will include an asset library, drag-and-drop functionality, and real-time previews of assets.

-Prototyping with Design Tools: Use Figma and Canva to create initial UI designs, focusing on usability and visual appeal. Leverage Adobe Illustrator and Inkscape for creating vector-based assets and UI elements.

- Interactive Visualization: Develop tools within the UI to allow users to see real-time changes to their assets and rendering settings, making it easier for artists and developers to iterate quickly.

4.Performance Optimizations

- Implement GPU optimizations through techniques like batching, instancing, and efficient texture management to reduce the number of draw calls.

- Use profiling tools to measure performance and identify bottlenecks, ensuring the rendering engine can handle complex scenes without

significant drops in frame rate.

- Optimize memory management for better handling of large textures and assets, reducing load times and enhancing the engine's responsiveness.

5. Testing and Debugging:

- Implement unit tests and manual testing of new rendering effects and UI features to ensure stability and reliability.

- Test across different platforms and hardware configurations to ensure compatibility and performance consistency.

6. Documentation and Tutorials:

- Write comprehensive documentation explaining the new rendering techniques, UI features, and how to integrate assets into the engine.

- Create step-by-step tutorials for artists and developers on using the new UI and working with advanced rendering effects.

- Include examples that demonstrate how the optimized pipeline and UI features can be used to create high-quality 3D scenes.

Timeline

Community Bonding Period (May 17 - June 12, 2025):

- Finalize the choice of rendering engine and familiarize myself with the existing codebase.

- Meet with the community to understand the needs of both developers and artists and gather feedback.

- Start designing the UI/UX elements in Figma and Canva, iterating on design ideas based on community input.

Phase 1: (June 13 - July 17, 2025):

- Begin implementing basic rendering pipeline enhancements (e.g., simple shaders, basic post-processing effects).

- Work on the asset management UI, integrating tools for handling textures, materials, and models.

- Set up initial performance profiling and begin optimizing the engine’s GPU usage.

Phase 2: (July 18 - August 21, 2025):

- Implement advanced rendering techniques such as PBR and integrate them into the rendering pipeline.

- Expand the asset management UI with features like real-time previews and drag-and-drop functionality.

- Implement more complex post-processing effects (e.g., depth of field, ambient occlusion, and motion blur).

Phase 3: (August 22 - September 6, 2025):

- Integrate real-time ray tracing for improved lighting, shadows, and reflections.

- Finalize the UI design, ensuring it is intuitive, responsive, and visually appealing.

- Conduct extensive testing and performance optimizations across multiple platforms.

Final Evaluation (September 7 - September 19, 2025):

- Finalize all changes and submit the code to the repository, ensuring all work is thoroughly tested and documented.

- Create and share comprehensive documentation and tutorials.

- Present a final report to mentors, summarizing the work completed during GSoC and the results achieved.

Why Me?

I bring a unique blend of technical expertise in computer graphics and practical experience in UI/UX design, making me an ideal candidate for this project. I have built my own game engine, where I implemented a variety of rendering algorithms, including PBR shaders, ray tracing, and post-processing effects. Additionally, I have hands-on experience with a wide range of design tools such as Canva, Figma, Adobe Illustrator, Inkscape, and Adobe Creative Cloud, allowing me to create intuitive and visually appealing interfaces.

My ability to combine rendering optimizations with effective UI/UX design will help me create a balanced and user-friendly rendering engine that caters to both developers and artists. I am excited about the opportunity to contribute to the open-source community by improving both the technical and visual aspects of rendering engines.

Project Goals

Develop a flexible and optimized rendering pipeline, incorporating advanced techniques like PBR and ray tracing.

- Design and integrate a user-friendly UI/UX for managing assets and visualizing rendered scenes.

- Ensure real-time performance with optimizations for both high-quality rendering and efficient resource management.

- Provide clear documentation and tutorials to guide users in utilizing the new features of the engine.

I am excited about the opportunity to contribute to this project and combine my skills in graphics programming and UI/UX design to create an improved, user-friendly rendering engine. Thank you for considering my proposal!