WebGL Water Tutorial - Rendering Pipeline Zustandsdiagramm WebGL Initialisieren - WebGL Kontext erstellen - VAO/Depth Erweiterungen laden Viewport einrichten Erfolgreich Assets Laden - Shader laden (Vertex/Fragment) - Shader Programme kompilieren - Texturen laden (dudv, normal, stein) - Meshes laden (water_plane, terrain) GPU Puffer erstellen Assets Geladen Framebuffer Einrichten - Reflektions Framebuffer erstellen - Refraktions Framebuffer erstellen - Tiefen Texturen einrichten Textur Parameter konfigurieren FB Bereit Render Schleife Wasser Rendern Wasser Shader Pipeline: - Wasser Shader Programm verwenden 1. Vertices in Clip Space transformieren - Wasser Mesh binden 2. Dudv Textur für Verzerrung sampeln - Perspektive/View Matrizen setzen - Reflektions Textur binden (TEXTURE1) 3. Reflektions/Refraktions Texturen sampeln - Refraktions Textur binden (TEXTURE0) 4. Fresnel Effekt anwenden - Dudv Textur binden (TEXTURE2) 5. Spiegelung Highlights berechnen - Normal Map binden (TEXTURE3) 6. Reflektion/Refraktion basierend auf Winkel mischen - Wasser Uniforms setzen (Reflektivität, Fresnel, etc.) 7. Mit Wasser Grundfarbe vermischen - Dudv Offset für Animation berechnen Wasser Ebenen Dreiecke zeichnen Nächster Frame Feltler/Beenden Echtzeit Updates: Status Aktualisieren - 60 FPS Render Schleife - Uhrzeit aktualisieren - WebSocket Status Sync - Kamera Matrix aktualisieren - Maus/Tastatur Eingabe - Wasser Eigenschaften aktualisieren - Kamera Orbit Steuerung Benutzereingaben verarbeiten Wasser Eigenschaften Änderungen Refraktion Rendern - Refraktions Framebuffer binden - Clipping Plane setzen (unter Wasser) - Terrain Mesh rendern Stein Textur anwenden Reflektion Rendern - Reflektions Framebuffer binden - Clipping Plane setzen (über Wasser) - View Matrix spiegeln - Terrain Mesh rendern Hauptszene Rendern - Haupt Framebuffer binden - Tiefen/Farb Puffer leeren - Wasseroberfläche rendern - Terrain rendern (kein Clipping) Multi-Texturing anwenden