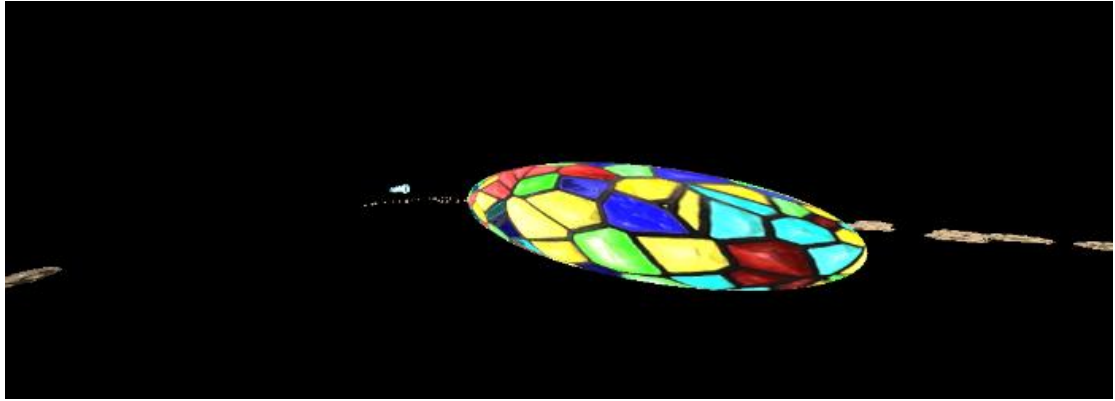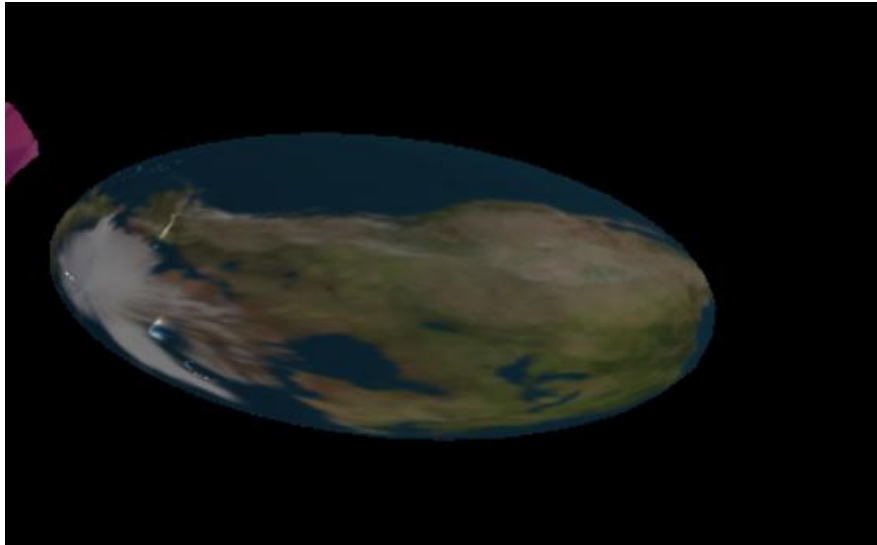Overall scene

spacescraft

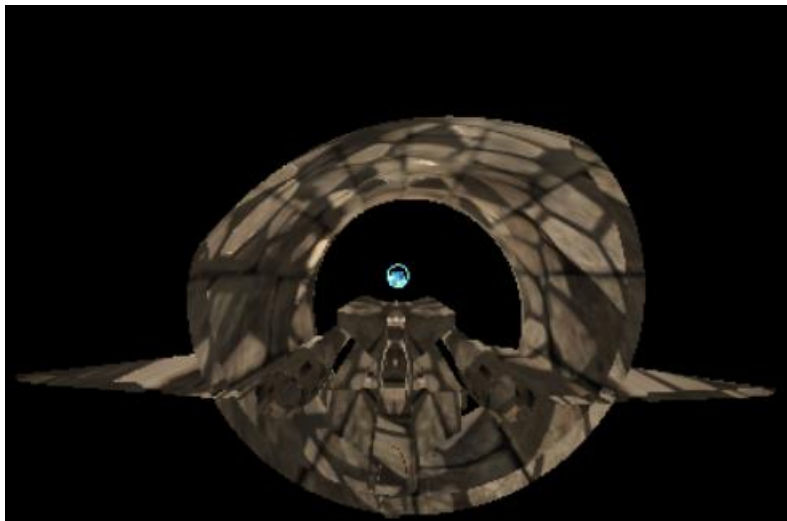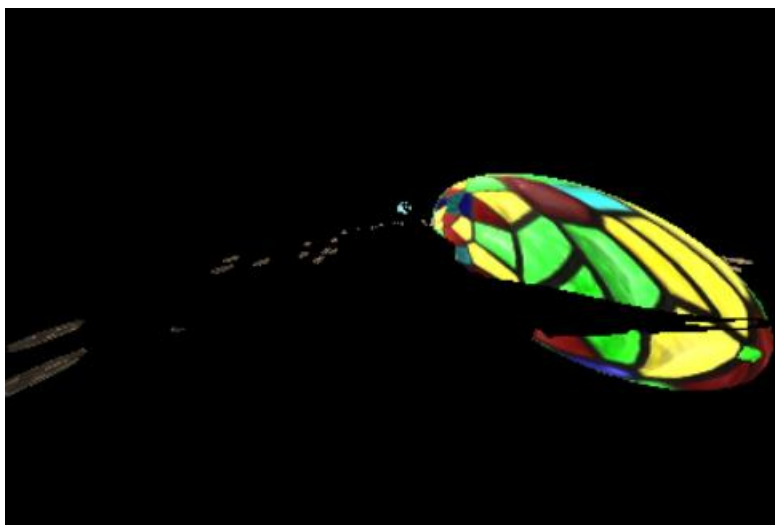wonderplanet

rocks

Energy ring

Earth

All above are close look at the basic light rendering

Visual feedback

Collision detection

```
{
    vec3 normal = normalize(normalWorld);
    if (normalMapping_flag)
    {
        normal = texture(myTextureSampler_normal, UV).rgb;
        normal = normalize(normal * 2.0 - 1.0);
    }
    vec3 lightVectorWorld = normalize(lightPositionWorld - vertexPositionWorld);
    float brightness = dot(lightVectorWorld, normal) * diffuseBrightness;
    vec3 diffuseLight = vec3 (brightness, brightness, brightness);

    vec3 reflectedLightVectorWorld = reflect(-lightVectorWorld, normal);
    vec3 eyeVectorWorld = normalize(eyePositionWorld - vertexPositionWorld);
    float s = clamp(dot(reflectedLightVectorWorld, eyeVectorWorld), 0, 1);
    s = pow(s, 50);
    vec3 specularLight = vec3(s * specularBrightness, s * specularBrightness, s * specularBrightness);

    //light source 2
    vec3 lightVectorWorld1 = normalize(lightPositionWorld1 - vertexPositionWorld);
    float brightness1 = dot(lightVectorWorld1, normal) * diffuseBrightness;
    vec3 diffuseLight1 = vec3 (brightness1, brightness1, brightness1);

    vec3 reflectedLightVectorWorld1 = reflect(-lightVectorWorld1, normal);
    float s1 = clamp(dot(reflectedLightVectorWorld1, eyeVectorWorld), 0, 1);
    s1 = pow(s1, 50);
    vec3 specularLight1 = vec3(s1 * specularBrightness, s1 * specularBrightness, s1 * specularBrightness);

    //spot light
    vec3 spotVectorWorld = normalize(spotPositionWorld - vertexPositionWorld);
    float spot = dot(spotVectorWorld, normal);
    vec3 spotLight = vec3(spot * spotBrightness, spot * spotBrightness, spot * spotBrightness);
    if (spot > cutoffangle)
```

More than one light source

Move the plane to get enough energy back to the earth.

----Use direction keys to control the plane.

---- q, w = tuning the lighting

---- move mouse to drive and rotate the plane