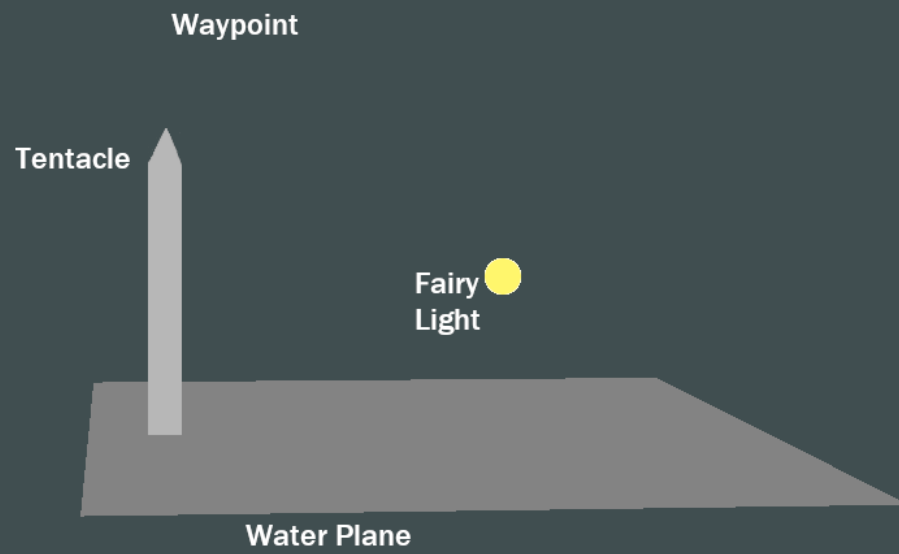# Fairies in Rain Scene

Second Draft

This final project demo will consist of a single scene, showing fireflies over water in rain, with a tentacle reaching out of the water and thrashing. The water is a simple plane whose vertices are displaced in a gerstner wave pattern. The water reflects objects above it using screen-space reflections. Above the water, there are "Fairies" which are simply lights, interpolating between waypoints using weighted curves. The fairies emit particles handled by a compute shader to produce a more interesting effect. Additionally, a compute shader calculates positions for rain particles. Finally, a tentacle constructed from a number of primitive cylinders/cones, extends out from the surface of the water, interpolating through a number of poses utilizing forward kinematics.
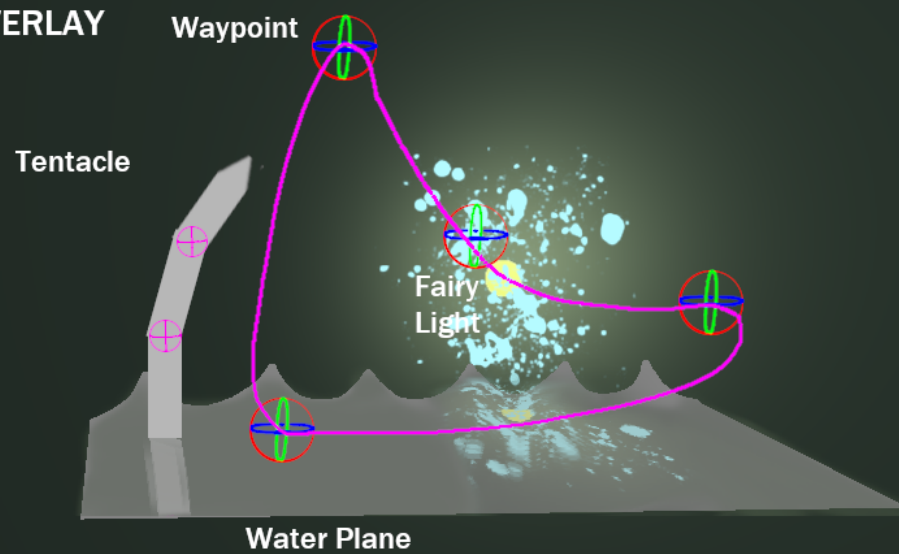
MOCKUP - BASE

Waypoint

Tentacle

Fairy
Light

Water Plane

MOCKUP - COMPLETE

Waypoint

Tentacle

Fairy
Light

Water Plane
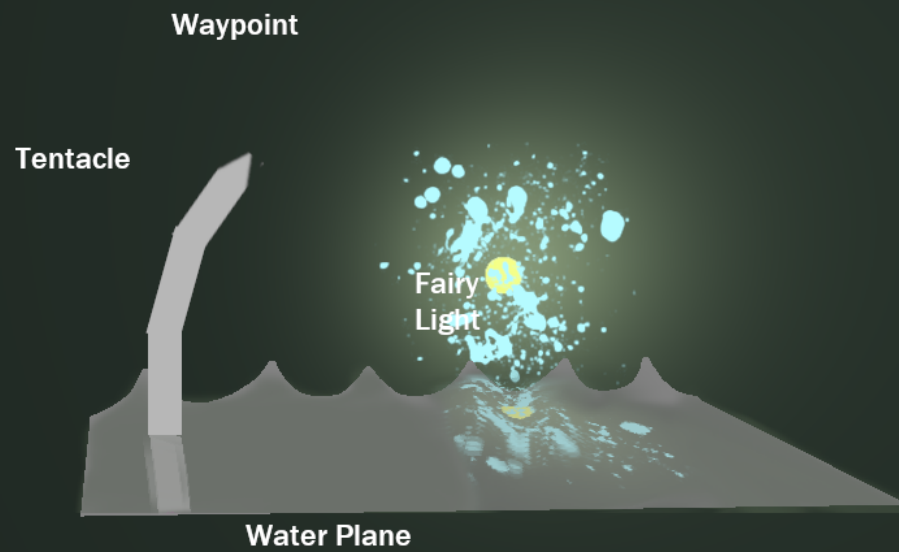
MOCKUP - COMPLETE
WITH OVERLAY

Waypoint

Tentacle

Fairy
Light

Water Plane

Project requirements:

- General shaders and pipelines; basic shader programs with vertex and fragment shaders; renderer architecture, components and design; fundamental lighting algorithms.
    - Phong shading, multi-light shaders, custom geometry, uniform buffers, etc.

  The demo, at present, is expected to use Phong shading, texturing, and vertex displacement algorithms for the water, clouds.

- Post-processing shaders and pipelines; framebuffers and multi-pass algorithms; intermediate lighting algorithms and effects.
    - Bloom, antialiasing, other post-processing effects; pipeline management systems; etc.

  Fireflies should utilize bloom.

- Screen-space effects; multiple render targets and intermediate framebuffer applications.
    - Deferred rendering, depth-of-field, motion blur, SSAO, etc.

  The water will use screen-space reflections to reflect objects above the water.

- Interpolation algorithms, curves and splines; intermediate shader programs with tessellation and geometry shaders.
    - Catmull-Rom, Bezier and Hermite curves; drawing and re-parameterizing curves; level-of-detail; etc.

  The displacement of the surface of water will follow mathematical curves, (Gerstner waves?)possibly with other noise added to break up the shape.

- Introductory animation programming concepts.
    - Forward kinematics and pose-to-pose keyframe animation; etc.

  A "tentacle" sticking out of the water that interpolates through a few poses, using forward kinematics.

- Advanced topics.
    - Compute shaders; scene culling; custom utilities; etc.

  Create additional fairy particle effect, calculating particle positions using a compute Shader. Also compute shader rain.