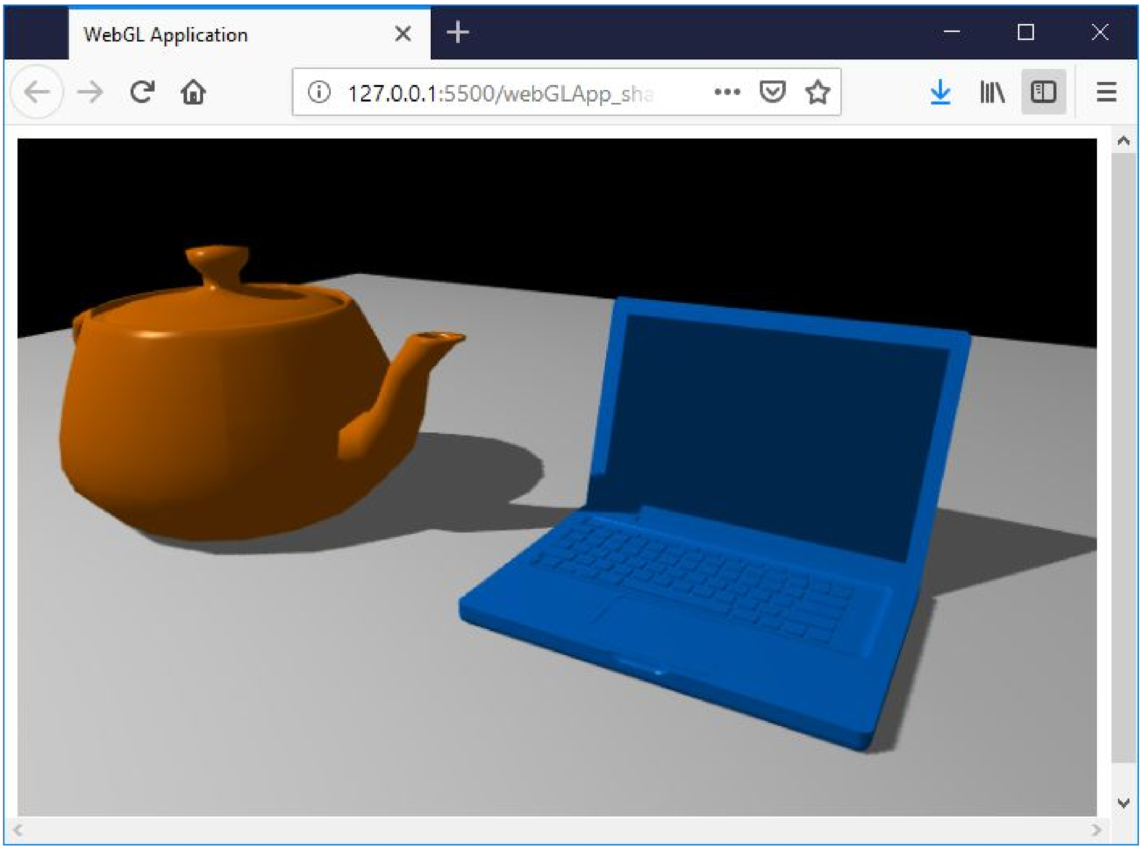
Augmented Reality

Project 2 (Computer Graphics) - May 2020

- If the code has errors, the exercise won’t be accepted for submission.
- Code is expected to be readable, clean, and optimal.
- To submit the exercise, upload a zip file named **“lastname1_name1_and_lastname2_name2.zip”** containing a folder with the .html file, the required javascript dependencies, and assets. Upload the file to the **“Project 2”** folder.
- No submissions will be accepted after **May 31st at 23:59:59**.



Statement

Extend the last exercise proposed at class (Phong shading) and implement the **shadow mapping** technique in order to add shadows to your scene. Your results should look similar to the image above.

You application should also include basic interaction using the mouse and the keyboard:

- Left mouse button: **rotation** around the scene.
- Mid mouse button: X and Y **panning**.
- Mouse wheel: **Zoom**-in and out.
- r key: **Reset** camera parameters (rotation, pan, and zoom).

Some considerations to take into account:

- Regarding the camera
    - Separate it to some distance about 5~10 units from the scene center.
    - Use perspective projection
    - Adjust the near and far planes to some distance about **0.1 (znear) and 20 (zfar) units** to contain the scene inside the view frustum.
- Regarding the light
    - It should be a directional light.
    - Separate it also about 10 units from the scene center.
    - Use an orthographic projection to simulate the light point of view projection.
        - Define the orthographic frustum with some values about -5, 5, -5, 5, 0, 30 (for left, right, bottom, top, near, and far planes).
- Regarding the algorithm, you will have two drawing functions:
    - *drawSceneFromLight()* that will render the scene from the point of view of the light.
        - It will render to a depth texture
        - It will use a shader that solely outputs pixel depths.
        - Needs to know model and light matrix transformations.
    - *drawSceneFromCamera()*
        - It will render to the screen
        - It will use a shader that performs phong shading and shadow mapping.
        - Needs to know model, camera, and light matrix transformations.

SOME ADVICES

Try to implement *drawSceneFromLight()* and *drawSceneFromCamera()* separately.

Test them individually to see that both are drawing the scene correctly (to test this, at first, don’t make *drawSceneFromLight()* render to a texture directly and make the shader output some plain color instead of the depth).

A second test could activate the render to texture (still outputting plain color) and draw a quad as a second pass to see if the texture contains the scene as seen from the light.

Finally, you should output the depth in the shader used by *drawSceneFromLight()* and use that output texture in the next pass to perform the shadow mapping.