

# Sebastian Greczek

Chicago, IL • 708-745-1475 • [sebastiangreczek1@gmail.com](mailto:sebastiangreczek1@gmail.com)

[GitHub](#) • [Portfolio Website](#) • [LinkedIn](#)

## CAREER OBJECTIVE

---

Graduating Computer Science major with a strong passion for software development seeking to start gain experience and grow as a developer.

## EDUCATION

---

### University of Illinois at Chicago (BS in Computer Science)

- Institution GPA: 3.75
- Expected Graduation: December 2021

### Selected Coursework

- Data Structures, Software Engineering, Database Systems, Computer Graphics, User Interface Design, Cloud Computing Objects, Development of Mobile Apps.

## SKILLS SUMMARY

---

- **Programming:** C/C++, Java, SQL, JavaScript, HTML/CSS, Python
- **Software Development:** Scrum Software Development(Agile), Jira, Git
- **Technologies/Frameworks:** Visual Studio, Eclipse, IntelliJ, JavaFX, UNIX, Windows

## PROJECT EXPERIENCE

---

### Javascript Raytracer

*Code:* [GitHub](#) *Live Demo:* [JSRaytracer](#)

- Developed a javascript raytracer that uses the Canvas API to draw six spheres to a html element. A path is traced through each pixel from the camera to the object. Once the ray reaches an object, the ray will bounce off the surface, towards the two light sources. Shadows are determined on whether the bounced off ray reaches the light sources or is intersected by another object first. The Blinn-Phong reflection model is used to determine the lighting. The user has control over the different lighting components, depth of reflections, and size, color, and position of the six spheres.

### Graph Algorithms Visualization

*Code:* [GitHub](#) *Live Demo:* [GraphAlgo](#)

- Creates a html grid that represents a graph data structure. The user can select cells on the grid to specify walls, set start cells, and set end cells. The user can then specify a traversal or search algorithm to simulate. A css animation will play showing how each cell is traversed based on the specified algorithm. If it is a search algorithm it will display a path that it found. The user can also add random weights to each cell which represent the cost of traversal to each cell. Mazes or walls can be randomly generated to show more interesting applications of graph algorithms.

### WebGL Shadow Mapping

*Code:* [GitHub](#) *Live Demo:* [ShadowMap](#)

- Developed a javascript program that uses the WebGL API to render a city skyline. The program takes in a string of coordinates and colors which are used to render the scene. The user can control the direction of the camera, change the perspective, change light direction, and zoom in and out. The user can also choose between a Chicago or Manhattan skyline. The shadow mapping technique is used to calculate the shadows. First the scene is rendered from the perspective of the light, the depth texture is extracted, and saved as a texture. Then the scene is rendered from the perspective of the camera and it is compared to the depth texture to determine whether the object is in shadow or light.