**Leon Frickensmith**

[leonfrickensmith@gmail.com](mailto:leonfrickensmith@gmail.com) github.com/lfricken 217 255 0626

| **EXPERIENCE**  Software Developer - Epic Systems - 2016 to Present   * Led development of a shared view framework used across several teams which included an asynchronous popup framework and new data reloading features * Spearheaded modernization effort using an internal framework to reduce development time by ~80 percent, saving hundreds of hours * Improved performance in core user workflows by 10 to 30 percent * Designed and developed full stack features across Referrals, Orders, and Medical Authorizations workflows * Fixed injection and other security holes in database * Analyzed legacy VB code to identify business logic and obfuscated behavior for consideration in our modernization effort * Debugged and fixed complex issues which only manifested in customer environments and backgrounded jobs * Visited hospitals to assist with customer installations and assess user needs and workflows * Implemented feature flags and security checks for Referrals   Software Developer Intern - Epic Systems - Summer 2015   * Co-created an MVC web app which used Google maps to help schedulers pick better locations for patient appointments | **EDUCATION**  BS Computer Science 2016  University of Illinois Urbana-Champaign   * Software Engineering I & II * Programming Languages and Compilers * Interactive Computer Graphics I & II * Artificial Intelligence * Database Systems * Numerical Methods   **TECH STACK**  C++ C# TypeScript Python Lua OpenGL React JavaScript Intersystems-Caché Git SVN  **INTERESTS**   * Distributed Computing * Physics and Simulations * Highly Testable Code * Software Design Principles * Programming Languages |
| --- | --- |

**OTHER PROJECTS**

| Illinois Robotics In Space   * Generated obstacle map by processing raw input from Xbox 360 Kinect's infrared sensor and accelerometer using libfreenect for use in NASA competition * Multithreaded Kinect processing and point-map transform operations to nearly double performance * Integrated obstacle information into the robot's path planning component via ROS Nodes | Space Combat MOBA   * Used libraries SFML and Box2D in C++ to create a real-time space combat videogame * Followed good design principles so multiplayer was simple to implement and debug * Organized and lead a team of 8 other devs in Software Engineering II to add features and tests as the primary semester project |
| --- | --- |
| Browser Based Boardgames   * Created Express based web app designed to be played by 5+ people * Utilized React on the client for good performance and responsive layout * Included Bootstrap to easily improve user experience, including a live chat and whisper system | GPU Accelerated Heat and Gas Simulation   * Designed a model of gas and heat flow on a 2D grid * Implemented OpenGL shader to multithread the computation to thousands of GPU processors * Created asynchronous API in C# to easily get, set, and change data within the simulation for use in a game |