Chetan Goenka

Berkeley, CA | (512) 363-2290 | cgoenka@berkeley.edu | github.com/c-goenka | linkedin.com/in/cgoenka

EDUCATION

University of California, Berkeley   Berkeley, CA
Master of Science in Electrical Engineering & Computer Science   May 2026

University of California, Berkeley   Berkeley, CA
Bachelor of Arts in Computer Science   Dec 2024
GPA: 3.90/4.0

SKILLS

Programming Languages: Python, Java, Javascript, SQL, C++, C, C#, HTML, CSS, Swift, Bash, Assembly, R, MATLAB
Frameworks/Tools: React, Node.js, Flask, Django, Pandas, NumPy, OpenAI, LangChain, Git, AWS, MongoDB, PostgreSQL

RELEVANT EXPERIENCE AND PROJECTS

BERKELEY INSTITUTE OF DESIGN - Human Computer Interaction Lab   Berkeley, CA
Engineering Design Scholar & Undergraduate Researcher   May 2024 – Jan 2025
- Engineered a web-based coding environment using React, CodeMirror, and OpenAI API, delivering dynamic, context-aware code explanations (line-by-line, block-wise, and data-flow) to enhance developer comprehension of AI-generated code
- Designed and implemented structured JSON metadata extraction with Zod, ensuring accurate line mappings, logical groupings, and data-flow tracking, enabling dynamically linked LLM-generated explanations and interactive overlays
- Extended CodeMirror's annotation capabilities by building custom DOM-based overlays for code elements, hover-triggered tooltips, and click-driven interactions, improving code navigation and readability through dynamic in-editor explanations

AI-POWERED BOARD GAME COMPANION - Board Wizard   Mar 2025 – Present
- Developed and deployed a Retrieval-Augmented Generation (RAG) web application using Python, LangChain, and OpenAI APIs that processes PDF rulebooks into vector embeddings to instantly answer board game rule questions
- Implemented a graph-based workflow using LangGraph's StateGraph to streamline the RAG process, combining vector similarity search with customized LLM prompting to generate contextually precise answers
- Built a responsive Streamlit user interface with chat functionality, game selection features, and efficient state management

PYTHON COMPILER   Jan 2024 – May 2024
- Developed a compiler using Java for ChocoPy, a statically typed dialect of Python, implementing key stages including lexical analysis, syntax analysis, and optimized code generation for RISC-V assembly
- Constructed a code generator to translate type-annotated ASTs into efficient RISC-V assembly code, integrating techniques such as constant propagation, dead code elimination, and register allocation, reducing runtime by 40%
- Implemented list operations, function calls, and control flow structures, ensuring correct execution of ChocoPy programs on a RISC-V simulator, and constructed an integrated semantic analyzer and type checker to maintain well-typed programs

PROCEDURAL WORLD GENERATION ENGINE   Jan 2023 – May 2023
- Devised and implemented a Java-based engine capable of generating and rendering pseudo-random, tile-based worlds with distinct rooms and hallways, allowing users to explore and interact with the environment through keyboard inputs
- Created and optimized complex algorithms for world generation, including the strategic placement, connection, and variation of rooms and hallways to ensure randomness, uniqueness, and diversity across multiple world instances
- Built a modular, scalable codebase using object-oriented principles, ensuring robustness using an extensive testing framework

ORACLE CORPORATION   Austin, TX
Data Science Intern   Jun 2022 – Aug 2022
- Evaluated 100+ marketing campaign responses using Python and SQL to extract key insights from customer interaction data
- Designed and implemented a predictive intelligence scoring system using Python, increasing marketing efficiency by 25%
- Enhanced data pre-processing and model deployment pipeline, decreasing overall run time by 18%