Benjamin Code

(630) 212-0327 - bjamescode@gmail.com - www.benjamincode.io - Chicago, IL

Career Objective

To work as a software engineer for a technology-first company where I can use and continuously improve upon my technical and problem solving skills to create and enhance real-time technology.

Qualification Highlights

- Masters Degree in Computer Science with a focus on Real Time Systems & Software Development
- Professional Experience in Sales, Marketing, Operations, and Creative roles
- Excellent problem solving skills in both technical and business domains
- Proven track record of exceeding expectations under strict timelines and extreme pressure

Education

DePaul University ……………………………………………………………………………… 2021 - 2024
Master of Science in Computer Science
Concentration: Real Time Systems & Software Development
GPA: 3.76/4.00 (Magna Cum Laude)

DePaul University ……………………………………………………………………….……… 2013 - 2017
Bachelor of Arts in Communication
GPA: 3.5/4.00 (Cum Laude)

Relevant Development Projects

(Additional information and demos can be found on my website, listed above)

Game Engine Development [C++, DirectX 11]
- Memory System
  - Developed heap-based memory system with individual allocation management capability
  - Overloaded New/Delete operators for heap assignment and alignment specification
  - Added tracking and debugging capabilities for capturing file and line number of each allocation
- Math System
  - Programmed a stable and comprehensive library capable of vector, matrix and quaternion operations
  - Improved library performance by a factor of 4 using SIMD Intrinsics (MMX, SSE)
- File System
  - Created Read/Write data storing mechanism using OS native functions for improved performance
  - Developed archiving system for packing and retrieving different data structures into and out of single files
- Object Hierarchy System
  - Implemented a tree-based data structure for handling hierarchies of objects in memory
  - Designed a system for implementing scene graphs and animation system in engine by propagating object transformations, lighting calculations, and various other properties from parent nodes to their children within any given animation
- Graphics System
  - *FBX Converter*
    - Programmed a converter to extract geometry and animation data from FBX files to reconstruct into a custom runtime file format
    - Created and used Google Protocol Buffers as intermediate format

- *Camera system*
  - ▫ Created a robust Camera System that supports multiple camera creations, camera management, view and projection matrix obtaining, frustum culling controls and more
  - ▫ Developed support for multiple camera views such as perspective and orthographic projection
- *Animation & Skinning system*
  - ▫ Wrote functions to load animation and skinning data of models from a custom resource file
  - ▫ Created a managed reserve pool of skeletons and meshes to be combined for animation instancing
  - ▫ Calculated animation and skinning matrices in real time based off of incoming geometry data
- *Shaders*
  - ▫ Wrote custom Pixel and Vertex HLSL shaders for lighting animations with customizable parameters
  - ▫ Wrote custom Compute Shaders for calculating animation keyframe interpolation to take advantage of GPU architecture
- *Vertex Buffers*
  - ▫ Programmed a D3D11 Vertex Buffer Manager enabling easy creation of Vertex Buffers with any combination of attributes
- *Collision Detection system*
  - ▫ Implemented basic collision detection using mathematically generated vertex and index buffers for basic shapes such as Ritter's Bounding Sphere

Game Performance Optimization [C++]
- ■ Reworked data structures to improve memory usage and data caching (10x speed boost)
- ■ Enhanced Vector and Matrix math library by using SIMD Intrinsics (5x speed boost)
- ■ Developed heap-based memory system for improved data management (12x speed boost)
- ■ Implemented Load-In-Place data structure for faster data initialization
- ■ Refactored and improved performance of a 200,000 particle system by 10x
  - Modified main loop to eliminate costly matrix multiplication
  - Cut down on memory usage by selecting more optimal data types
  - Utilized RVO to avoid unnecessary temporary constructions
  - Designed project-specific memory scheme to reduce and group memory allocations

Multithreaded Audio Engines [C++]
- ■ XAudio2 Engine
  - Used the XAudio2 API to to extract, process, and play raw audio within a custom game engine using 7 separate manager threads (Main thread, Resource loading thread, Audio Voice threads, Playback Control threads, Voice Coordination threads, Error Handling thread, Kill thread)
  - Properly protected all game and audio resources via system handles and safely used them with XAudio2 callbacks
  - Created support for complex playlist creation, volume controls, track blending, panning, time controls, and pitch controls
- ■ Windows API Engine
  - Developed a real-time multithreaded audio engine with 6 manager threads and 20+ subordinate worker threads
  - Created a custom Win32 Handle System incorporating mutexes, futures, promises, async calls, and conditional variables to protect engine resources shared between threads
  - Developed a memory safe, inter-thread communication system by implementing the Command design pattern with independent thread queues to accept messages

Modern Architecture Space Invaders Remake [C#]
- Recreated the classic arcade game *Space Invaders* using 13 object oriented design patterns and 140+ classes
- Developed support for animating and moving textured sprites, processing keyboard input, detecting multiple types of collisions, a font system, responsive audio, levels/scenes, etc.
- Employed Object Pooling of game sprites to reduce calls to the New Operator and improve performance
- Implemented an early-out collision system that drastically cut down on collision checks

MS Paint Clone [Java]
- Created a simple paint app with basic mouse and keyboard input reading, color/shape/brush selection options, fill options, etc.
- Used a myriad of design patterns to create a well designed system with sound architecture

Distributed Blockchain System [Java]
- Created a secure, SHA-256 hashed, proof-of-work blockchain distributed over a number of servers
- Implemented a peer-to-peer, multicast and broadcast distribution system for sharing the blockchain ledger

Todo App [JavaScript, HTML/CSS, Node.js, MongoDB]
- Created a Todo React CRUD app with user profiles and login authentication

Professional Experience

Account Executive, Trigger Group, 2020-2021
- Joined Trigger Group as a part of their first US team, growing their presence and business in North America as DoorDash's outsourced Sales team
- Conducted research for business development and customer growth
- Pitched products and services to potential customers while managing existing relationships
- Created informational decks to assist in training new team members

NBCUniversal Umbrella, 2017-2020
- **Project Coordinator, Peacock (NBCUniversal Digital Enterprises)**, 2019-2020
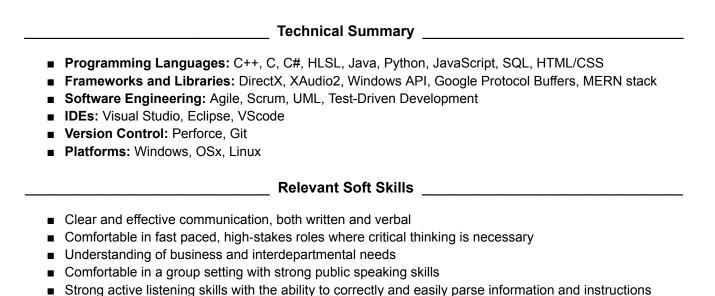 - Joined launch team for NBCU's streaming platform, Peacock, and continued previous responsibilities in addition to the following
  - Created pitch decks for Peacock platform design and direction
  - Created presentation decks for Digital Enterprises upper management
  - Worked with Data Analytics teams to shape platform development and focus

- **Project Coordinator, Snapchat (NBCU x Snap Joint Venture)**, 2018 - 2019
  - Internally recruited from NBCUniversal to launch their scripted content studio after NBCU's 500mm investment in Snap, inc.
  - Created pitch decks for creative marketing and short form content
  - Performed research and managed special projects for the partnership management, business development, research, and legal teams
  - Acted as main point of contact for technical & design needs with external vendors and contractors

- **Page, NBCU Page Program**, 2017 - 2018
  - Highly selective rotational leadership development program with positions in Universal Pictures Brand Marketing, NBC Alternative Development, and Universal Studios Hollywood Operations
  - Developed a strong foundation in business leadership, people management, and creative direction

Technical Summary

- **Programming Languages:** C++, C, C#, HLSL, Java, Python, JavaScript, SQL, HTML/CSS
- **Frameworks and Libraries:** DirectX, XAudio2, Windows API, Google Protocol Buffers, MERN stack
- **Software Engineering:** Agile, Scrum, UML, Test-Driven Development
- **IDEs:** Visual Studio, Eclipse, VScode
- **Version Control:** Perforce, Git
- **Platforms:** Windows, OSx, Linux

Relevant Soft Skills

- Clear and effective communication, both written and verbal
- Comfortable in fast paced, high-stakes roles where critical thinking is necessary
- Understanding of business and interdepartmental needs
- Comfortable in a group setting with strong public speaking skills
- Strong active listening skills with the ability to correctly and easily parse information and instructions

Extras

- Always happy to lose at chess
- Could build the chess set
- A feline father of two

- Can name at least 6 dog breeds
- Cooks a mean chicken dinner
- Code really is my last name