Cheng Jin

C++|C#|JAVA|Python|Unity|Anti-Cheat Expert, Problem Solver

Looking for an opportunity to satisfy clients with professional and earnest work.

Available for any time zone

Email: gbapyou@gmail.com

A freelancer & fast learner with full experience in software development for over 7 years, especially good at C++/C# and easily getting used to a new technical environment. Having worked as a freelancer on a lot of projects for over 7 years, I accumulated many experiences in my skills like data structures, algorithms, mathematics and cryptography. If you are looking for a talented software developer then you won't be disappointed with my work. But if you only hire people who have "done it before" then you will miss out on every marginalized person like me. Now I am looking for an opportunity to satisfy clients with my professional and earnest work. I am willing to prove myself through any kind of test courses. Thank you.

TECHNICAL SKILL

No	Skill	Period (Years)
1	C++	11
2	C#	11
3	Java	7
4	Python	3
5	PHP	2
6	Visual Studio	11
7	Unity	11
8	Android Studio, Eclipse	7
9	WDK (Windows Driver Kit)	3
10	WinDgb (Windows Kernel Debugger)	2
11	Windows App Development	7
12	Android App Development	7
13	Android NDK	7
14	Windows Kernel Module Development	2
15	Reverse Engineering	2
16	Linux App Development	1
17	OpenSSL, cURL	2
18	OpenCV	3
19	FFmpeg	1
20	Assimp	2
21	Google Mediapipe	1
22	OBS Studio	1
23	Cryptography	
24	Mathematics	

MAJOR PROJECTS

1. **Anti-Cheat System (C++, C#, PHP, cURL, OpenSSL)**
   https://github.com/nxtboyIII/fairplay

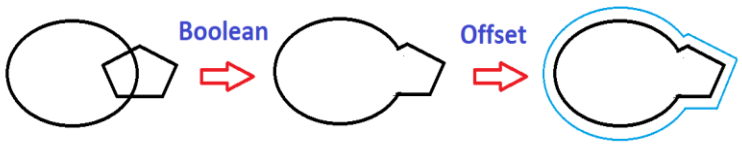
   A system that protects the mobile game from any kind of hacks. The system has been applied for a game called “BulletForce”, which is online multiplayer shooting game that you can easily find in Google Play store. Many kinds of illegal products had been up and around the internet in order to ruin ecosystem of that game, before my anti-cheat system was released. But now there has never been a single solitary hacks that broke into my system for a few months. I designed a very sophisticated structure in order to confuse the hackers. Also, I customized Photon server client SDK in order to encrypt all data being transferred between players. I used Android NDK (C++ module) for client system, and PHP for the backend system. Several cryptographic algorithms like MD5, SHA has been used for secure communication and user validation.

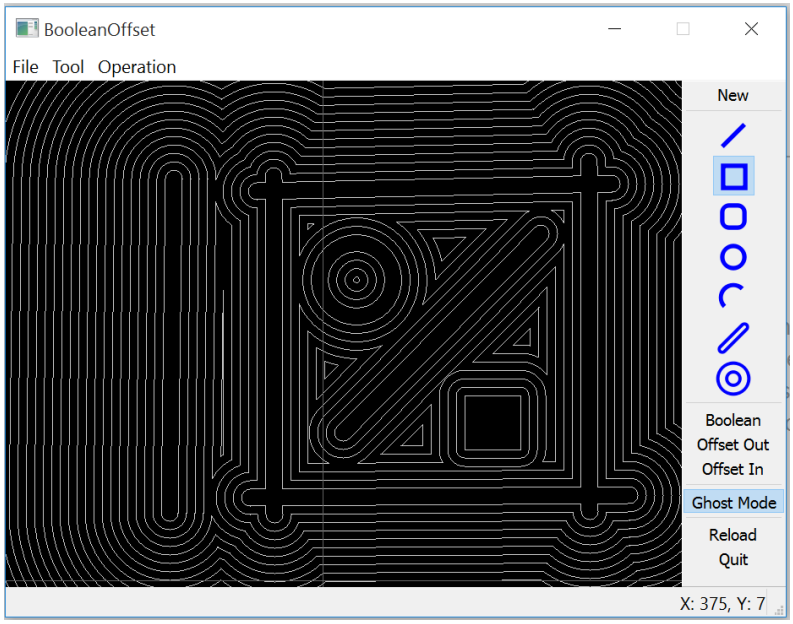
2. **Data & Anti-Theft Protection System (C++, ASM, VHDL, WinDGB)**

   A system that protects the data from being stolen. This project is quite similar to TrueCrypt and VeraCrypt, but not exactly the same, because the data is protected in hardware level. We used FPGA as a core firmware device that contains AES-256 algorithm in order to encrypt/decrypt all data being transferred between hard drive and memory. The main board was designed to use PCI-X interface and I implemented AES-256 algorithm by using 4,500 logical elements inside the FPGA.
   The system works for both of Windows and Linux. I was in charge of Windows Device Driver (.sys) and VHDL module.

3. **Offset/Boolean Operation CAD Engine (C++, QT, Visual Studio, Assimp).**
   https://github.com/Gbapy/Offset-Boolean-CAD

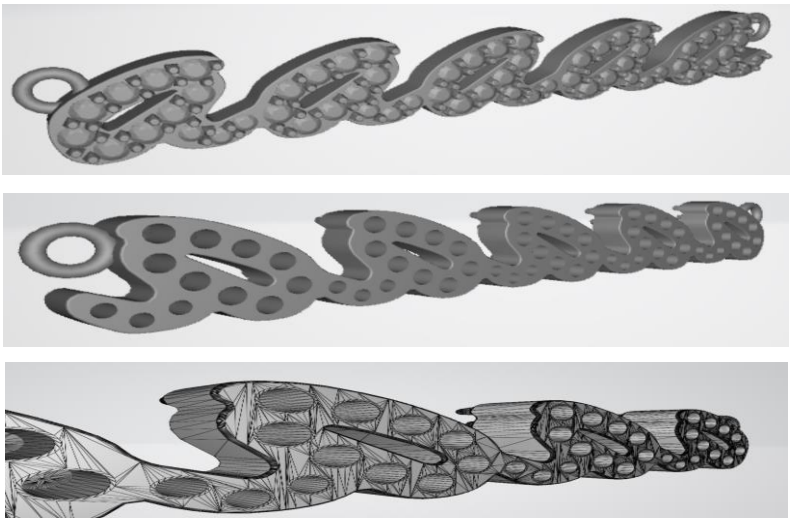
   A high-performance CAD engine that performs Offset/Boolean operation, which is very useful for electronic circuit board design. This project has been developed in C++ (Visual Studio & QT) with a lot of advanced data structures and algorithms. The biggest problem of this project was to consider a proper tolerance when performing double-type data calculations.





4. **3D Model Generating Engine with GLB/STL export for 3D print (C++, Python).**
   https://github.com/Gbapy/3D-Model-Generate-Engine-C-Python-

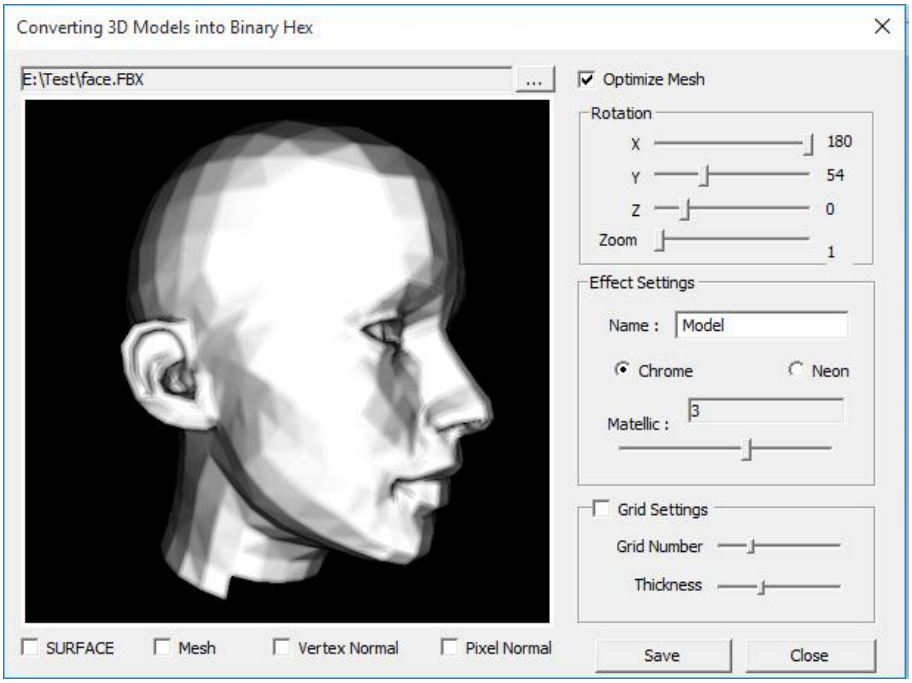
A high-performance engine that is used in backend in order to dynamically generate 3D models of several accessories like rings, necklaces according to user's online input.



This engine is very high-performance. It takes fewer than 50 milliseconds to generate a model, even for the big model with vast number of vertices. I used C++ for the core engine (Linux shared object) and Python as a wrapper to let PHP call C++ engine. This engine supports GLB 2.0/STL export that I developed myself.

5. **3D Rendering with Several Effects (C++, Assimp)**
   https://github.com/Gbapy/3D-Rendering-Project-without-using-any-DirectX-OpenGL

   An engine that renders 3D models and generates a couple of effects like chrome, neon. In order to prove my 3D mathematics skill, I didn't use any built-in libraries like DirectX, OpenGL. This project has been implemented only by C++ with a lot of mathematical solutions and advanced high-performance algorithms.

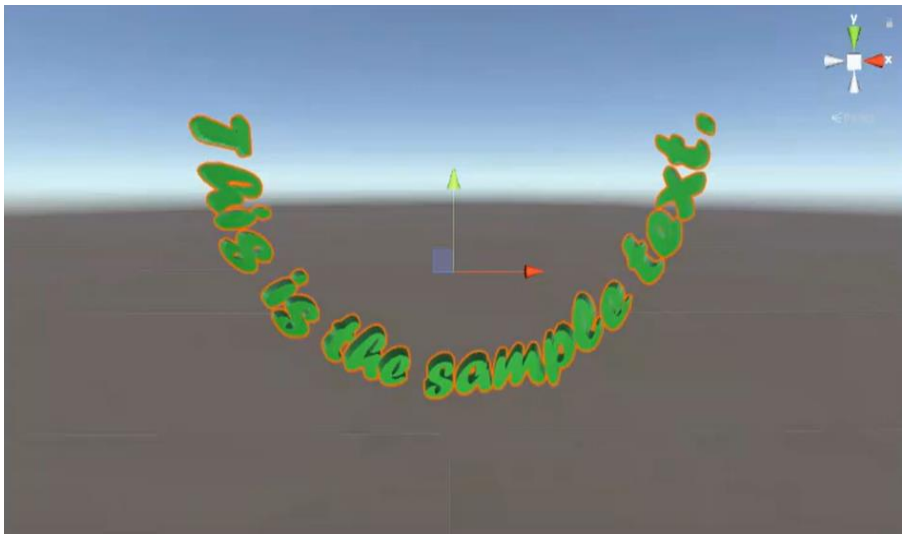


6. **3D Modeling Plugin for Unity (C++, C#, Unity, Assimp)**
   https://youtu.be/DyM0rXeAc7k

   A plugin for Unity that is used for 3D modeling. An advantage of this plugin is that you can easily generate 3D models from 2D images without having knowledge about tricky things like vertices, meshes, etc.

7. **Font Plugin for Unity (C++, C#, Unity, Assimp)**
   https://github.com/Gbapy/Font-Plugin-for-Unity
   https://youtu.be/MV9FSutD8Xo

   A plugin for Unity that allows developer to easily generate 3D text models with several fonts ever installed in OS, which makes it easier to handle fonts that is one of very annoying parts of Unity.



8. **WeChat bot (C++, Reverse Engineering, IDA pro)**
   https://github.com/Gbapy/Wechat-chatbot-using-reverse-engineering

   An engine that hooks WeChat module and automatically sends messages (including images) at a certain interval. I used DLL injecting and reverse engineering technologies.

9. **Nox App Player Recording Application (C++, Reverse Engineering, FFmpeg)**
   https://github.com/Gbapy/NoxPlayer-Recorder-by-hooking-the-DirectX-OpenGL
   https://github.com/Gbapy/Multiple-Webcam-Recoder-with-audio-using-FFMPEG-h264-

   It is a little difficult to record Nox screen, because DirectX/OpenGL renders the screen of Nox App Player. So, I used reverse engineering and DLL injecting technologies in order to hook the DirectX/OpenGL module inside the Nox. Also, I used FFMPEG in order to generate video files using h264 codec. I referred to the OBS Studio that is a well-organized open source. (https://github.com/obsproject/obs-studio)

LinkedIn: **https://www.linkedin.com/in/liang-zhai-a95a83190/**

GitHub: **https://github.com/Gbapy**

Email: **gbapyou@gmail.com**