

RYAN BIEKER

SOFTWARE ENGINEER

rbieker@proton.me

ryanbieker.com

linkedin.com/in/ryan-bieker

SUMMARY

I'm a Software Engineer building applications and developing games in Unreal Engine. Passionate in gameplay programming and game design. I also work in web development as well. Away from the keyboard I compete in fencing and Esports tournaments, participate in Game Jams, work in 3D Modeling, and study Japanese.

EDUCATION

Texas State University

2022 - 2025

Bachelor's Degree in Computer Science
Minor in Applied Mathematics

Game Developer Society
Google Developer Student Club
Fencing Club - Saber Team Captain

LANGUAGES

C++
C#
C
Java
Python
JavaScript
TypeScript
HTML
CSS

SOFTWARE AND TOOLS

Visual Studio
Visual Studio Code
Unreal Engine 5
Blender
Git
GitHub
Azure DevOps

SOFT SKILLS

Self-Motivated
Strong Work Ethic
Problem Solving
Interpersonal Communication
Attention to Detail

PROFESSIONAL EXPERIENCE

Software Engineer

Scala | 2024 - 2025

- Developed websites in JavaScript, HTML, and CSS for clients' websites while communicating with a team of 3 UX/UI designers in Figma, Adobe Illustrator, and Miro.
- Refactored JavaScript for optimization resulting in a 25% improvement to site loading time and responsiveness.
- Early hire on to the startup team of 4 built on creativity and problem solving.

PROJECT EXPERIENCE

Gameplay Programmer | Game Designer | QA Tester

Bleak Contact | 2024

- Developed an interactive story using Unreal Engine 5 and Blender with C++ and Blueprints with 3 releases.
- Released the game resulting in over 17,000+ impressions and reached the New and Popular tag for itch.io.
- Built 5 robust, maintainable, extensible systems with version control in Git through Azure DevOps.
- Game design documentation, world building, narrative design, and level design with whiteboxing for 3 levels.
- User testing with 15+ individuals for

Gameplay Programmer | Game Designer | QA Tester

War Card Game | 2024

- Created a competitive 2 person game in Java against an AI based on War.
- Designed game mechanics through object-oriented programming with over 3 million possible card configurations.
- Utilized agile methodologies with version control in Git and documentation in 13 pages of Javadoc.
- Led code reviews in a team of 3 developers with weekly meetings with testing of player engagement.