

# Eric Delmonico

contact@ericdelmonico.com | <https://github.com/EricDelmonico> | [www.ericdelmonico.com](http://www.ericdelmonico.com)

## Education

**Bachelor of Science**, Rochester Institute of Technology (RIT), Rochester, NY

Graduated December 2022

- *Major*: Game Design and Development | GPA: 3.90
- *Minor*: Computer Science, Japanese

## Work Experience

**Unity Developer**, Viomerse, Remote (May 2023-Present)

- Leading development of XR medical training application using Unity, C#, WPF, and C++ to add business value to Viomerse's augmented reality headset; achieved savings of \$150,000/yr with software
- Built multimedia layout authoring tool in Unity game engine using C#, HLSL pixel/compute shaders, and Netcode for GameObjects to add functionality to AR Headset
- Programmed C# Windows Service to track USB devices in order to detect proprietary hardware and automatically start Viomerse's multimedia layout authoring tool to improve ease of use for the end user
- Facilitated inter-process communication using Windows Named Pipes/ServiceWire to interface between multiple applications with different .NET runtimes to make the software more modular

**Nintendo Switch Port Dev/Unity C# Programmer**, *Idol Hell (PC and Switch)*, Rochester NY (June 2022 - Nov. 2024)

- Spearheaded development of Nintendo Switch port using a hardware developer's kit and Nintendo's Unity SDK; funded by successful \$17,500 Kickstarter campaign
- Added new UI features such as a scene transition system and high scores menu to improve polish
- Wrote new gameplay systems such as boss companions in order to add variety and interest
- Implemented a save system utilizing fixed-sized structs to optimize save file size on PC and Switch
- Wrote HLSL shader code and used Shader Graph for in-game effects such as making bullets vibrate when they are being grazed in order to better telegraph gameplay features and give feedback to players
- Optimized game to run at 60 FPS on the Nintendo Switch Console to ensure smooth gameplay
- Authored tool to auto-convert all scenes/prefabs to use Addressables/AssetBundles and manually swapped project to async workflow to eliminate stuttering on the Nintendo Switch and optimize patch size

**Contract Software Engineer**, Summit Technology Services, Remote (June 2021 - Aug. 2021)

- Designed and implemented noise reduction UI for product software using C# and Google protobuf
- Prototyped free trial system for product features using SQL, C#, and .NET Entity Framework
- Diagnosed and repaired bugs across the codebase using C# to facilitate a smooth product launch

**Software Engineer Intern**, Waygate Technologies, Remote (June 2020 - Nov. 2020)

- Architected UI shell for video playback on a handset using C# and WPF port (NoesisGUI)
- Adapted remote connection system from previous handset generation to current generation using C#, WPF/XAML, MVVM design pattern, and NuGet
- Navigated Agile software development environment through Microsoft Team Foundation Server/Azure DevOps and attended virtual scrums in Microsoft Teams
- Maintained existing code by ensuring compliance to Dependency Injection design pattern

## Featured Projects

**Looking for Beans** - RGDC Game Jam at RIT, Won "Best Design"

- Implemented laser and gun mechanics, camera controls, and other core mechanics using C# / 3D math
- Redesigned levels and refined core mechanics to create a polished final project

**Wallbreakers** - Jam for a Cause Game Jam at RIT, Won "Best in Show"

- Assembled and directed design of scene traversal system using C# and Unity; designed and developed smooth visual transition system between scenes
- Programmed core pattern-matching game mechanic using C# and Unity

**Mirror Maze** - RGDC Game Jam at RIT, Won "Best Overall"

- Assisted in design of core mechanic, collaborated on sound design and recording
- Programmed and tuned ambient sound/footstep systems to improve user experience

## Skills

**Technologies:** Unity, Visual Studio, Git, WPF, DirectX 11, DirectX 12, WebGL, RenderDoc

**Languages:** C#, C++, C, JavaScript, HLSL, HTML/CSS, SQL, Python