

Contact

r1ngegamedev@proton.me

www.linkedin.com/in/r1nge
(LinkedIn)
r1nge.com (Blog)

Top Skills

Python (Programming Language)
Git
C#

Languages

Russian (Native or Bilingual)
English (Professional Working)

Certifications

EFSET English Certificate 71/100
(C2 Advanced)

Nikita Kruglik

Unity Developer 5+ YOE TG: @R1nge
Russia

Summary

6 years in gamedev; 10+ years in programming.
Transitioning into DevOps, but still have interest in game development
Contacts:
<https://t.me/r1nge>

Blog:
<https://r1nge.com>

Projects:
The web of fate (WIP)
<https://game.r1nge.com>
- Jobs burst
- Unsafe
- Flood fill algo
- Mask shader
- Render texture
1000x faster; Sub-ms optimization (0-1 ms) on mobile wasm

Suika with modding support
- VContainer UniTask DoTween
- New input system, Touch/Mouse/Keyboard/Gamepad support
- Vibration, gamepad rumble
- UI navigation using Keyboard/Gamepad/Mouse/Touch
- UI dark/light theme
- Modding support
- Collider generation based on the sprite
- Sprites caching, runtime compression
- Music player, audio streaming
- Merge sound priority system
- Runtime mod validation
- Multiple game modes
- Save/Load last game
- Combo system
- High scores

Windows Android WebGL
<https://r1nge.itch.io/moddable-suika>
<https://github.com/R1nge/Suika>
<https://r1nge.github.io/>

Bomberman

- Real-time multiplayer
- Procedural map generation
- Character skins
- In-game powerups
- VFX Pooling

<https://github.com/R1nge/BomberMan>

Hyper Casual game

- VContainer, UniTask, DoTween
- Addressables
- Localization
- Notifications
- Runtime navmesh baking
- VFX Pooling

<https://github.com/R1nge/RunAndBomb>

<https://r1nge.itch.io/runandbomb>

AI based 3D virtual anime companion (wAIfu)

- Voice recognition
- Text generation
- Translation
- Voice synthesis
- Lip sync
- VRM to Unity auto import

Unity frontend

Python backend for Nns

ZeroMQ for data transfer using TCP sockets

Progress (Top – newest)

<https://youtu.be/wlIBgkIWuyI>

<https://youtu.be/Nu91NxKmh-k>

<https://youtu.be/KBZrHFgAmYs?si=0AQRcruAmrM9nIyf>

<https://github.com/R1nge/OpenWaifu>

Spiders - performance test

Ecs vs MB Jobs Burst

<https://github.com/R1nge/Spiders>

Game for the GMTK game jam

<https://r1nge.itch.io/battladventure>

DOTS

<https://github.com/R1nge/DOTS>

Fixed a bug in a popular API

<https://github.com/Animechan-API/animechan/issues/99>

Sysadmin posts

Securing internet facing services

<https://r1nge.com/7-securing-internet-facing-services.html>

Nginx authentication

<https://r1nge.com/10-securing-internet-facing-service-follow-up.html>

Ubuntu lts 22 to 24 to 26

<https://r1nge.com/11-ubuntu-lts-upgrade.html>

TCP fastopen

<https://r1nge.com/12-tcp-fast-open-nginx.html>

Nginx RCE + Brotli

<https://r1nge.com/13-nginx-rce.html>

Nginx security headers

<https://r1nge.com/14-improving-nginx-config.html>

Apparmor Nginx

<https://r1nge.com/15-apparmor-hardening.html>

More posts on IaC, automation, ci/cd and docker security to come!

Gamedev posts

Telegram Vtuber Avatar

<https://r1nge.com/6-telegram-avatar.html>

Performance comparison between ECS and MB + Burst

<https://r1nge.com/4-spiders-jobs-vs-morpeh-ecs.html>

Multiplayer bomberman + csp

<https://r1nge.com/3-multiplayer-bomberman.html>

Wolves voice control

<https://r1nge.com/2-wolves-voice-control.html>

Runtime navmesh generation

<https://r1nge.com/1-runtime-navmesh-backing-for-ai-navigation.html>

Moddable game

<https://r1nge.com/0-how-vtubers-inspired-me-to-create-a-moddable-game.html>

2025

Telegram avatar

Upon receiving a video it would feed it into a virtual camera hooked to VTubeStudio, speed up the video, record the screen

Slow down the recording and send it back;

Also have a support for switching emotions via SharpHook

<https://r1nge.com/6-telegram-avatar.html>

Wolves' emperor

Use a voice command to trigger tamed wolves attack a target

(java, python, named pipe; windows/linux)

<https://r1nge.com/2-wolves-voice-control.html>

I was in need for a color picker that works with UI Toolkit;

None of the solutions worked on the latest Unity version, so I rolled my own;

Rolling dice bot (Telegram, c++, arduino, c# backend)

I've constructed the frame out of a shoe box, taped some rgb leds, dumped an arduino and DC motor in here.

The cabling was a mess, 3 different inputs;

Could've done a better job at this buying a couple of dc to dc converters

Started digging into self-hosting (Docker, nginx, kernel security hardening, k8s)

2024

Attempted to launch a tank.io game (protobuf, vanilla js canvas, go backend)

Too much dev time's required;

I would've a much better time doing this using Unity instead

2021-2023

Lots of unfinished projects;

Most of them gamedev related, some of them to learn, others to make a buck.

2020

Developed an external aim bot based on real-time image recognition for cs:go

Used the same model trained on a different dataset (coco) to automate labeling

SoftInterMob LLC

Senior Unity Developer

April 2025 - June 2025 (3 months)

DOTS ECS, R3, VContainer, Message pipe, UI Toolkit

Винторог

Unity Developer

July 2024 - April 2025 (10 months)

Russia

100K DAU

As the main Unity developer on the project, I was able to:

Reduce build size, loading times, RAM usage by adding Addressables package

Refactor screen logic to simplify the process of adding new ones

Fix Telegram caching issue

Fix missing 'application/wasm' mime type on DigitalOcean

Make UI adaptable to Telegram fullscreen mode

Add seasonal events (Halloween, X-mas)

Localize UI using Unity's Localization package

WebGL, UniRx, Zenject, Jslib, REST

Anrum

Unity Developer

May 2024 - July 2024 (3 months)

Krasnodar, Russia

Was developing hyper-casual games as an individual contractor and was able to quickly prototype core mechanics, implement animations, vfx and UI for the games, connect sdks and publish to the store.

WebGL, DoTween, AdMob, YG sdk, I2 Localization, Photon Quantum, Jslib

<https://yandex.ru/games/app/294859?draft=true&lang=en>

<https://yandex.ru/games/app/257688?draft=true&lang=en>

Butcher's games

Unity Developer

December 2023 - May 2024 (6 months)

Mostly asset flips or decompilation using external tools. Implemented core gameplay and UI

Android, DoTween, New Input system, Shader graph

[https://play.google.com/store/apps/details?](https://play.google.com/store/apps/details?id=com.butchersgames.sniperhero3d)

[id=com.butchersgames.sniperhero3d](https://play.google.com/store/apps/details?id=com.butchersgames.sniperhero3d)

<https://play.google.com/store/apps/details?id=com.smash.car.crash.simulator>

<https://play.google.com/store/apps/details?id=com.mega.ramp.jump.car>

Freelance (Self employed)

Unity Developer

January 2020 - December 2023 (4 years)

Russia

During that time I was tinkering with the various topics, some of which are AR, modding, multiplayer, physics, ECS, and more. Basically, I was able to touch every aspect of Unity in some way or another, except for VR.

Created AR experiences using Vuforia

Worked with VRM models

Implemented asynchronous web requests using UniTasks (REST)

Optimized RAM usage and build size using Addressables

Used dependency injection frameworks (Zenject, VContainer)

Translated with the help of Unity's localization package

Implemented player authentication and data storage using PlayFab

Developed real-time multiplayer projects using Netcode for GameObjects and hosted games using Unity's relay

Used various plugins (DoTween, Cinemachine, Odin inspector, TriInspector, A* pathfinding)

Created captivating visual effects using Shader Graph

Configured physics, ragdolls, and car physics

Worked with particle system, animations (blend trees, layers), Firebase,

Docker, js, Go and Protobuf

Contributed to the Morpeh ECS repo