

# SAI NARAYAN NATARAJAN

Technical Artist and Programmer

(614) 966 – 1389

sainarayan3@gmail.com

www.linkedin.com/in/sainarayan3

www.sainarayan.me

@nightmask3

## TECHNICAL SKILLS:

- Game Engines: Skilled in the use of Unreal 4 and Unreal 3, basic knowledge of Unity
- Languages: C++, C, C#, Blueprints/Visual scripting, Python, MaxScript, HLSL, GLSL
- Technical Art tools: 3D Studio Max, Houdini, Photoshop, Blender, Simplygon, xNormal
- Technical Art specializations: VFX, shaders, character rigging and animation, environment assets
- Gameplay Programming: Character movement, animation and input systems, procedural generation
- Physics Programming: Fluid simulation, rigid body physics, constraint solvers
- AI Programming: Behavior trees, planning, pathfinding, navigation and movement
- Source control: Perforce, Git, SVN

## PROFESSIONAL EXPERIENCE:

**Technical Artist, 'Forza Motorsport 7'**  
Turn 10 Studios, Redmond

Aug 2018 – Ongoing

- 'Forza Motorsport 7' is a simulation racing game in the proprietary Forza engine for Xbox One and PC
- Upgraded studio-wide LOD tooling from Simplygon 5 to Simplygon 8, batch modified over 30,000 model assets to work with changes to the Simplygon API and verified no loss of functionality or performance targets
- Worked with cinematics director, character lead and technical animators to define technical needs and improvements for the character pipeline in preparation for the next Forza title
- Unblocked production artists with issues that crop up in their workflow in the day-to-day
- Wrote technical documentation for the character pipeline and proposals for tooling improvements

**Tools Programmer and Technical Artist, 'Bebylon Battle Royale'**  
Kite & Lightning, Remote work

Jan – May 2018

- 'Bebylon Battle Royale' is a VR fighting game in Unreal 4 for PC and PS4
- Created tools for artists using Python and C++ such as rigs for animating, build tools, generic property manipulators, animation retiming and curve extraction tools, data validation tools, static asset analysis tools
- Debugged UI, networking and engine issues in the Unreal codebase with > 2 million lines of code

**Gameplay Programmer and Technical Artist, 'Starship Commander'**  
Human Interact, Remote work

May – Nov 2017

- 'Starship Commander' is a VR sci-fi action game in Unreal 4 for PC and PS4
- Implemented enemy AI for combat behaviors, flight systems using splines for movement
- Created art tools like asteroid field spawners, designer tweakable electric/lightning effects
- Wrote shaders and created art assets for player attacks & shields VFX, UI elements for targeting systems

## PROJECT EXPERIENCE:

**Lead Programmer and Technical Artist, 'Project Gilgamesh'**  
Team Kitteh, Redmond

May 2017 – Ongoing

- 'Project Gilgamesh' is a cinematic platformer and adventure game made using Unreal Engine 4 for PC
- Wrote a volumetric fog shader using HLSL and Unreal's node based shader system
- Created a custom character movement system centered around parkour-based movement
- Implemented a gameplay system for interactive elements and materials for things like grass catching fire or metal conducting electricity inspired by Legend of Zelda: Breath of the Wild
- Modelled, rigged and animated the main character and added cloth physics to enhance movement visuals

**Physics Programmer, '3D Physics Engine'** Jan – May 2017  
DigiPen Institute of Technology, Redmond

- Implemented a 3D physics engine using Observer pattern, Entity-Components, and ImGui for UI
- Collision detection with Gilbert-Johnson-Keerthy (GJK) method and Expanding Polytope Algorithm (EPA)
- Collision resolution with Sequential Impulse LCP solver and constraints

**Lead Programmer and Technical Artist, 'The Village'** May 2016 – May 2017  
DigiPen Institute of Technology, Redmond

- Led a team of 7 to create a third person exploration game using Unreal 4, was the sole artist
- Implemented systems of player movement, player interaction, quests, AI behaviors, and scripted events
- Created all art assets for environmental/characters/prop models, made all Materials, animations and VFX

**Physics and Gameplay Programmer, Level Designer, 'Unlit'** Jan – May 2016  
DigiPen Institute of Technology, Redmond

- 'Unlit' was a 2.5D sidescrolling platformer made in a C++ engine from scratch, with three other programmers
- Built a physics engine with colliders for cubes, spheres, planes, raycasting and RK4 integration
- Implemented systems for input from keyboard and gamepad, tested and tuned player input for gameplay

**Physics Programmer, 'Fluid Simulation'** April – July 2016  
DigiPen Institute of Technology, Redmond

- Implemented fluid simulation using 2D smoothed particle hydrodynamics (SPH)
- Wrote a plugin for proprietary engine that integrated OpenCL to enable computation on the GPU
- Wrote an API to allow for any calling application to utilize OpenCL in a generic way

**Tools Programmer, 'Perlin Heightmap Builder'** Aug – Dec 2015  
SRM University, Chennai

- A C++ project to create a tool that would allow a user to generate heightmaps for terrain using Perlin Noise
- The user could view the output of their creation and modify it, and then export their output as heightmaps

**Level Designer and Producer, 'The Nightmare'** Feb – July 2014  
HAN University, Arnhem

- 'The Nightmare' was a first person horror platformer made using the Unreal Development Kit/Unreal 3.
- Coordinated a team of 20 people using a combination of Waterfall and Agile development methodologies
- Designed the concept for the game and oversaw creative direction

## **EDUCATION:**

**Master of Science in Computer Science** May 2017  
DigiPen Institute of Technology, Redmond, Washington

**Bachelor in Computer Science** May 2015  
SRM University, Chennai, India