# Simulacrum Labs — Client User Guide

Version 2.9.21 | April 2026

---

This guide walks you through using the Simulacrum platform to run simulation training jobs, generate ML annotations, execute hyperparameter sweeps, and run Docker-only GPU workloads on distributed consumer GPUs.

## Getting Started

### Create Your Account

- Go to **app.simulacrumlabs.com**
- Sign up with your email, Google account, or GitHub account.
- If you are part of a team, one person creates an Organization and invites others. All team members share the same credit balance and job history.

### Add Credits

- Navigate to the **Billing** section after signing in.
- Purchase a credit pack: **$25** (~33 GPU-hours), **$100** (~133 GPU-hours), or **$250** (~333 GPU-hours).
- All pricing is flat at **$0.75 per GPU-hour**. No hidden fees, no egress charges.
- Credits are deducted in real-time while your job runs.
- Free trial: **$5 free credit** on signup, no credit card required.

## Job Modes

Simulacrum supports four job modes. Choose the one that fits your workload:

### 1. Annotations (Rendering + ML Data)

Renders your UE5 scene and generates ML training data. Upload your UE5 project, set the number of frames, and submit. You receive:

- RGB camera images
- Depth maps
- Semantic segmentation masks (Cityscapes-compatible)
- Instance segmentation masks
- 2D and 3D bounding boxes
- LiDAR point clouds (.pcd files)
- Camera calibration data (intrinsics + extrinsics)

- Object tracking IDs
- Label maps (JSON)
- COCO-format annotation manifest

## 2. UE5 + Python Training (Run-as-Is)

Runs your Python training script against a live UE5 simulation environment. Your script connects to AirSim/Colosseum inside the simulation and trains a reinforcement learning agent. When complete, you download the trained model files (PyTorch .pth, .onnx, .zip).

**Your training script should:**

- Accept hyperparameters via **argparse** (for sweep support)
- Read the **AIRSIM_HOST** environment variable for the AirSim connection address
- Save trained model files to the current working directory
- Include a **requirements.txt** with all Python dependencies

## 3. Parameter Sweeps

Toggle 'Sweep Mode' in the job submission form. Define parameter names and values (e.g., learning_rate: 0.0003, 0.001). The platform generates all combinations and dispatches each variant to a separate GPU node simultaneously. When all variants complete, download all results and compare.

## 4. Docker Only (No UE5 Required)

Run any GPU workload without Unreal Engine. Upload a zip containing your Python code and a requirements.txt. The platform runs everything inside a Docker container with full GPU passthrough. Perfect for:

- **NVIDIA IsaacSim** — robotics simulation and training
- **Gazebo** — ROS-based robot simulation
- **PyBullet** — physics simulation for RL
- Custom PyTorch/TensorFlow training pipelines
- Any Docker-based GPU workload

**How to submit a Docker Only job:**

- Select **Docker Only** in the job mode selector
- Upload a .zip or .7z containing your Python files and requirements.txt
- Specify the Python script path (defaults to **main.py** if not set)
- No .uproject file needed — the zip only needs your code

# Preparing Your Project

## For UE5 jobs (Annotations, Training, Sweeps):

Your project zip should contain:

- A **.uproject** file (Unreal Engine 5.7 project)
- All project content (Content/, Config/, Plugins/, etc.)
- For training jobs: Python training scripts and a **requirements.txt**
- Upload formats: **.zip** or **.7z** (7z compresses 30-70% smaller)

**Tip:** Your zip can contain the project at root level or inside one subfolder. The platform searches up to 2 levels deep for the .uproject file automatically.

## For Docker Only jobs:

- A Python script (e.g., main.py or train.py)
- A **requirements.txt** listing all Python dependencies
- Any data files or configuration your script needs
- No .uproject file needed

# Upload and Configure

- Click **'Upload Project'** and select your .zip or .7z file. Files over 100MB upload directly to Azure (SAS upload).
- Once uploaded, the platform scans your zip and populates dropdowns for Map Path, Python Script, Vehicle Mode, and Control Mode.
- For Docker Only jobs, only the Python script path field is shown.

# Security

- **gVisor Sandbox:** Optional — check 'Require gVisor Sandbox' before submitting. Your job only runs on nodes with gVisor kernel-level container isolation. Unchecked by default for maximum node availability.
- **Secure Wipe:** All project data, frames, and intermediate files are automatically overwritten with zeros and deleted after every job.
- **Docker Isolation:** Every job runs in an isolated Docker container. Client code cannot access the operator's personal files.
- **US-Only Compute:** All nodes verified as US-based via IP geolocation.

# Monitoring Jobs

- Active jobs appear in the job list with real-time status updates.

- Training jobs show live screenshot previews from inside the simulation (updated every 10 seconds).
- Sweeps show per-variant progress in the Active Sweeps section.
- Use **'Complete Job'** to gracefully stop a training job and collect results.
- Use **'Cancel Job'** to abort (partial results may not be saved).

## Downloading Results

- **Annotation jobs:** Click 'Download Frames' to get all annotation files as a zip.
- **Training jobs:** Click 'Download Results' to get trained model files.
- **Sweeps:** Click 'Download All Results' to get results from all variants.
- **Docker Only:** Click 'Download Results' to get all output files from the container.

## Pricing

**Flat rate: $0.75 per GPU-hour.** No tiers, no multipliers, no egress fees.

| Credit Pack | GPU-Hours | Per-Hour Rate |
|---|---|---|
| $25 | ~33 hours | $0.75/hr |
| $100 | ~133 hours | $0.75/hr |
| $250 | ~333 hours | $0.75/hr |
| Enterprise | Custom | Monthly invoicing |

Credits deduct in real-time while your job runs. If a job fails, you are not charged.

# API Key Management

API keys allow you to submit and manage jobs programmatically without using the web dashboard.

**Creating Keys:**

- Go to **Settings > API Keys** on the dashboard.
- Click **Create API Key**.
- Copy the key immediately — it is only shown once.

Include the key in your requests as:

```
Authorization: Bearer YOUR_API_KEY
```

**Revoking Keys:**

Click the revoke button next to any key in Settings. Revoked keys stop working immediately. Active jobs submitted with a revoked key continue to completion.

# Team Accounts (Organizations)

Simulacrum supports team accounts through Clerk Organizations. One person creates the organization, invites team members via email, and everyone shares the same credit balance, job history, results, and uploaded projects.

- **Admin:** Manages billing and members.
- **Member:** Submits jobs and views results.

# Supported Frameworks

- Unreal Engine 5.7 (for UE5 job modes)
- AirSim / Colosseum (built-in, auto-configured)
- Python 3.12 with pip (runs inside Docker container)
- PyTorch (pre-installed, CUDA 12.8, GPU enabled)
- Stable Baselines3 (PPO, SAC, A2C, etc.)
- Gymnasium, NumPy, OpenCV, and standard ML libraries
- Any Python package installable via pip (specify in requirements.txt)

# Troubleshooting

**Upload fails or times out**

Files over 100MB use SAS direct upload. If it still fails, check your internet connection and try again. Both .zip and .7z formats are supported.

**Job fails immediately**

For UE5 jobs: check that your zip contains a valid .uproject file (can be at root or one subfolder deep). For Docker Only jobs: ensure your Python script path is correct and requirements.txt is present.

**Training script doesn't connect to AirSim**

Make sure your script reads the AIRSIM_HOST environment variable: AIRSIM_HOST = os.environ.get('AIRSIM_HOST', 'localhost'). Connect to that host on port 41451. The platform auto-configures AirSim — you do not need to set up settings.json.

**Job completed but no output files**

The platform scans for model files (.pth, .onnx, .zip, .pkl) in the working directory and standard output folders. Make sure your script saves output files to the current working directory or a subfolder like output/, results/, or trained_model/.

---

**Need Help?**

Email: help@simulacrumlabs.com

Website: simulacrumlabs.com