LOW POLY

war pack

by polyperfect

Have a Suggestion?

info@polyperfect.com

Thanks!

First of all, thank you for purchasing our pack, we really appreciate that! We are putting a lot of effort into this.

We are also planning to expand the list of the characters and their animations in the future with free updates of the pack. Check out our **Facebook Page** for any news.

Handy Links ;)

Other Low Poly Packs

Low Poly Animated Animals

Low Poly Animated People

Low Poly Animated Dinosaurs

Low Poly Animated Prehistoric Animals

Low Poly Ultimate Pack

Low Poly War Pack

2D Packs

Low Poly Icon Pack

Fancy Icon Pack

Follow us

Facebook

Instagram

Youtube

Polyperfect.com

Updates

VERSION 2.0

New scripts, new models, new characters. New everything!

We have remade the pack from the ground up. All the scripts, animations, rigs, and models were changed so please, DO NOT UPDATE from the older versions. Also is worth mentioning that there will be probably another smaller update to balance and polish things out. We hope you will like our work, we have put LOT effort and love into this small pack.

VERSION 1.25

Bug fixes and tweaks

Player

Player Controller

If you refer to the Demo scene we have created for you, you will see there is both a player and a bunch of enemies for you to shoot at. If you click on the player in the scene, you will see that attached is the Player controller Script. (Please be aware, the whole movement script for this character at the moment is using Root motion, there are some in place animations included in the project for you to play with if you do not want to use this movement method and want to create a character controller yourself. A character controller using RigidBody or a character controller is currently in development for the next release.

If you do not know what root motion is please refer to some online tutorials as it can get rather complex or message us and we will try our best to answer any questions you might have.

How it works

At the top of the script, you will see a couple of following variables you can change if you like:

Health

This is the health variable of the Player

Speed

This is so that you can make the player walk faster or slower.

Running Speed

This is so that you can make the player run faster or slower.

Move Smoothing

This will determine the smoothes of the character transition between idle/walking/running.

Turn Smoothing

This will determine the smoothes of the character turning towards the camera direction.

Weapons

The weapon has been created to make it easy to change some values depending on how you want them to be, how much damage you want each gun to produce, the range of each gun, and more.

Weapon flash

This is the Gameobject we are using to make a flash appear at the end of the weapon.

Weapon size

Setting this higher will give you the option to add a new gun (If you have some you want to add).

Inside each of the Weapons will be some more Variables that are used to control some core features.

Weapon Name

This is used to trigger the correct gun aiming animation in the animator.

Weapon GameObject

This is the actual GameObject located under the right wrist in the characters hierarchy.

Switch out time

This is how long it will take for the current weapon to be turned off, and the new one turned on.

Weapon end

This is a GameObject located at the end of each of the gun, this is used to set the GunFlash animation position for the weapon.

Left-hand position

This is the GameObject that is located inside each gun and is the position that the left hand of the character will IK to.

Weapon damage

This is how much damage this weapon will cause to the enemy's

Weapon range

This is how far you will be able to hit characters from when aiming for this weapon.

Switch weapon key

You can choose which Key you want to press to swap to this weapon.

Aiming Variables

These variables have been made public so that if you feel the aiming of one of the guns is off, press play and adjust these values until you think that it looks as expected, then right-click on the script and click Copy Component. Then when out of play mode, right-click on the script once more and paste values.

Camera Setup

Camera

This is the camera you are using inside the soldier, this can be located in the soldier extras component which holds all the necessary parts to make the camera movement and move are achievable.

Caperaxspinner

This is the Gameobject located beneath SoldierCamSetUp, this is to rotate the camera on its X-Axis to allow proper rotation of the character.

Ik Set Up

IK

This is a Boolean, turn this on if you want
to use Ik on the character, turn it off if you don't. (Turning this off may cause some strange behavior of animations so better to keep it on)

IK aim pos

This is the Gameobject that is located in front of the character, it is called Ik Head Aim. This is the Gameobject you can drag around the scene and make the head look during play mode.

Spine

This is the Transform located inside the characters Hierarchy of bones that
will rotate when the camera rotates, changing this will cause some very strange behavior and will require you to change all the Aiming Variables (Located in the Gun Set-Up) to match the new spine rotation.

Other

Aiming point offset

This is the point on an invisible ray going from the center of the camera which the spine of the character will always rotate towards. This is how we have made sure to get the character looking where the center of the Camera is looking. (Changing this value will make the white sphere move forward and backward to visualize where the spine is aiming at)

Keys

W - forward

S - back

A - left

D - right

E - interaction

SHIFT - sprit

RIGHT MOUSE - weapon aim

LEFT MOUSE - shoot

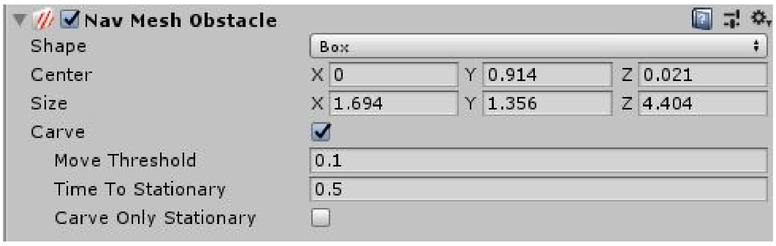
1 , 2 , 3 , 4 , 5 , 6 , 7 - switch weapon (setup able)

Enemy

Enemies are based on the Unity Navigation system. Each enemy has an Enemy script and NavMeshAgent attached to them to be able to move.

Setup scene

First, all objects that you want the enemies to walk on you need to set as Navigation Static. Second of all, objects that will be moving in run time (cars, tanks, etc.) need to have a NavMeshObstacle component on them. Set it like this (Of course dimensions of the obstacle set accordingly):



Nav Mesh Obstacle ☑

Shape: Box

Center: X 0 Y 0.914 Z 0.021

Size: X 1.694 Y 1.356 Z 4.404

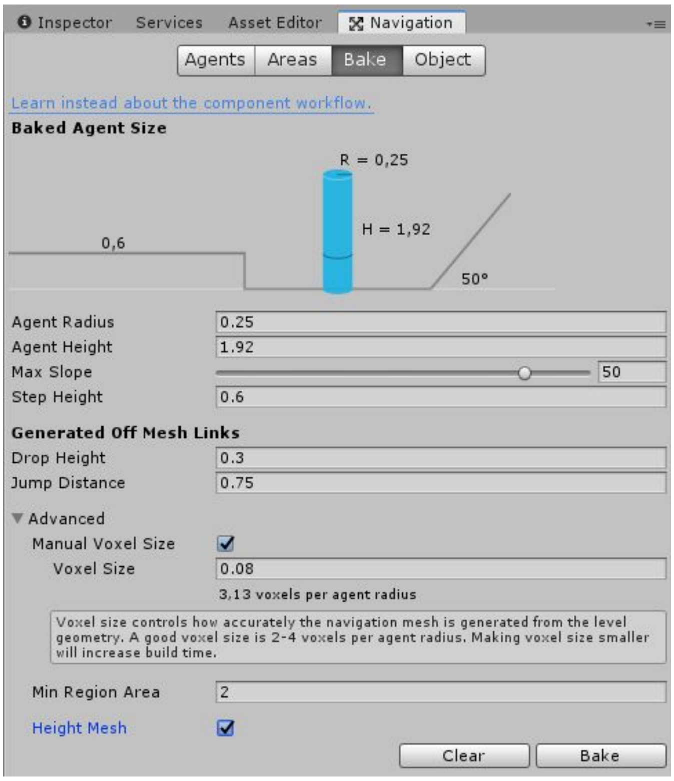
Carve: ☑

Move Threshold: 0.1

Time To Stationary: 0.5

Carve Only Stationary: □

And then you need to bake the NavMesh of your scene. Open the Navigation window and in the bake tab set values and click Bake. These are our recommended values:



Baked Agent Size

Agent Radius: 0.25

Agent Height: 1.92

Max Slope: 50

Step Height: 0.6

Generated Off Mesh Links

Drop Height: 0.3

Jump Distance: 0.75

Advanced

Manual Voxel Size: ☑

Voxel Size: 0.08

3.13 voxels per agent radius

Voxel size controls how accurately the navigation mesh is generated from the level geometry. A good voxel size is 2-4 voxels per agent radius. Making voxel size smaller will increase build time.

Min Region Area: 2

Height Mesh: ☑

Enemy Script

The script is using unity NavMeshAgent to control enemies.

Health

Amount of enemy health points.

Canvas

This is the canvas that will be shown on top of the enemy.

Health Slider

This is a UI Slider which shows enemy health.

NavPath

The path that the enemy will walk on. More about NavPaths here.

Eyes Pos

Transform of enemy eyes. From this position form which enemy will spot the player.

View Range

Float, which determines the maximum distance (in meters) from which the enemy will visually spot the player.

Max Shoot Range

Float, which determines the maximum distance (in meters) from which the enemy can shoot targets.

Time To Lose Aggro

Time in seconds in which the enemy stops looking for the player after he loses him.

Difficulty

This determines the damage that the enemy does. 1 is 100% of original weapon damage, 0.5 is 50% of original weapon damage, and so on.

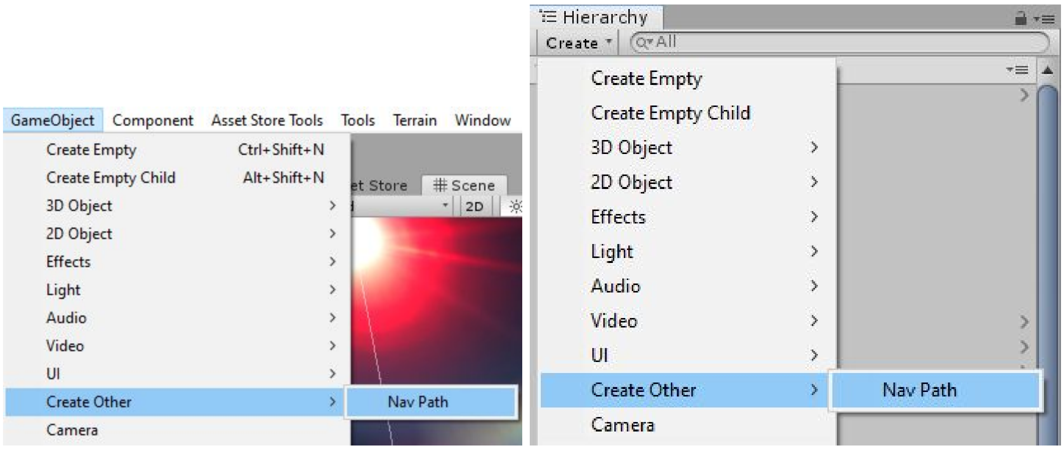
Scatter

Scatter of weapons. The higher the value, the less accurate the enemy will shoot.

Weapons

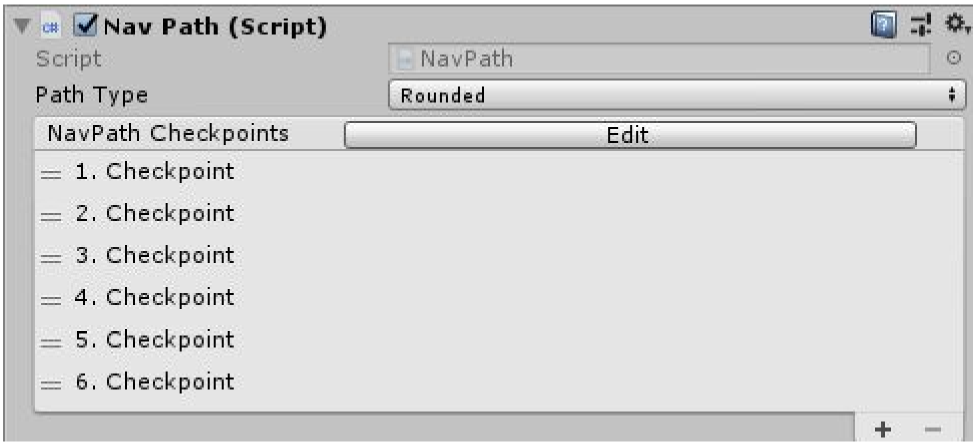
NavPath

NavPath is a GameObject which stores path checkpoints through which the enemy will walk. You can create NavPath in the GameObject menu by clicking Create Other/Nav Path or in Hierarchy by clicking Create/Create Other/Nav Path.



NavPath Script

Now let's talk about how to create and edit the path.



Nav Path (Script)

Script: NavPath

Path Type: Rounded

NavPath Checkpoints

1. Checkpoint
2. Checkpoint
3. Checkpoint
4. Checkpoint
5. Checkpoint
6. Checkpoint

Path Type

Here you can select from three types of paths:

- Rounded - Enemy will walk in the circuit, next checkpoint after last is first.

- One way - Enemy will walk towards the last checkpoint and then stay there.
- Both ways - Enemy will be forever walking from first to the last checkpoint and then back to the start.

NavPath Checkpoints

On top of all checkpoints is the edit toggle button which allows you to edit checkpoints. If the edit toggle is on and you select any of the checkpoints you can change its position in the scene window.

Vehicles

All vehicles consist of three main parts:

- Vehicle model with controller script
- Canvas
- Camera script

Each vehicle controller is separated into two tabs Setup and Statistics.

Car Controller

The car controller is based on Unity wheel colliders.

Set up

The first tab is Set up and here you can set up car wheels, lights, and other essential variables.

Wheels

Here you can determine the number of axles in a size field. In each axle, you must select the right and left wheel collider.

Motor

This is a Boolean, turn this on if you want this axle to be connected to the engine.

Steering

This is a Boolean, turn this on if you want this axle to be able to turn.

Brake

Turn this Boolean on if you want this axle to have brakes.

Handbrake

Turn this Boolean on if you want this axle to be connected to the handbrake.

Lights

Lights are divided into front and backlights. For each, you can add multiple unity lights.

Front Lights Renderer

This is a link to the renderer of front lights mesh.

Front Lights On Material

This is material that will be used on the Front Lights Renderer when you turn lights on.

Front Lights Off Material

This is material that will be used on the Front Lights Renderer when you turn lights off.

Back Lights Renderer

This is a link to the renderer of backlights mesh.

Back Lights On Material

This is material that will be used on the Back Lights Renderer when you turn lights on.

Back Lights Off Material

This is material that will be used on the Back Lights Renderer when you turn lights off.

User Interface

Like every vehicle, it has a Health UI foldout in which you can choose the type of health UI (Text, Slider, Text, and Slider).

Dependent on what you choose you will see those fields:

Text

This is TextMeshPro which shows health as text.

Slider

This is a Slider that shows health as a linear bar.

Speed Text

This is TextMeshPro which shows car speed as text.

Interaction Setup

Canvas

This is the canvas that will be shown when driving a car.

Player Slot

Position of the player while driving a car.

Get Out Point

Position of player relative to the car when you exit.

Blow Up FX

GameObject prefab that will be instantiated on car destroy.

Virtual Camera

Virtual Camera that will be switched to when you enter the car.

Stop Interacting Key

You can choose which Key you want to press to exit the car.

Statistics

In the second tab are variables that change statistics and characteristics of the car.

Health

Amount of car health points.

Motor Torque

This is torque in Newton-meters that will be applied on all axes with Motor boolean set to true.

Brake Power

This is torque in Newton-meters that will be applied as brake power on all axes with Brake boolean set to true.

Max Speed

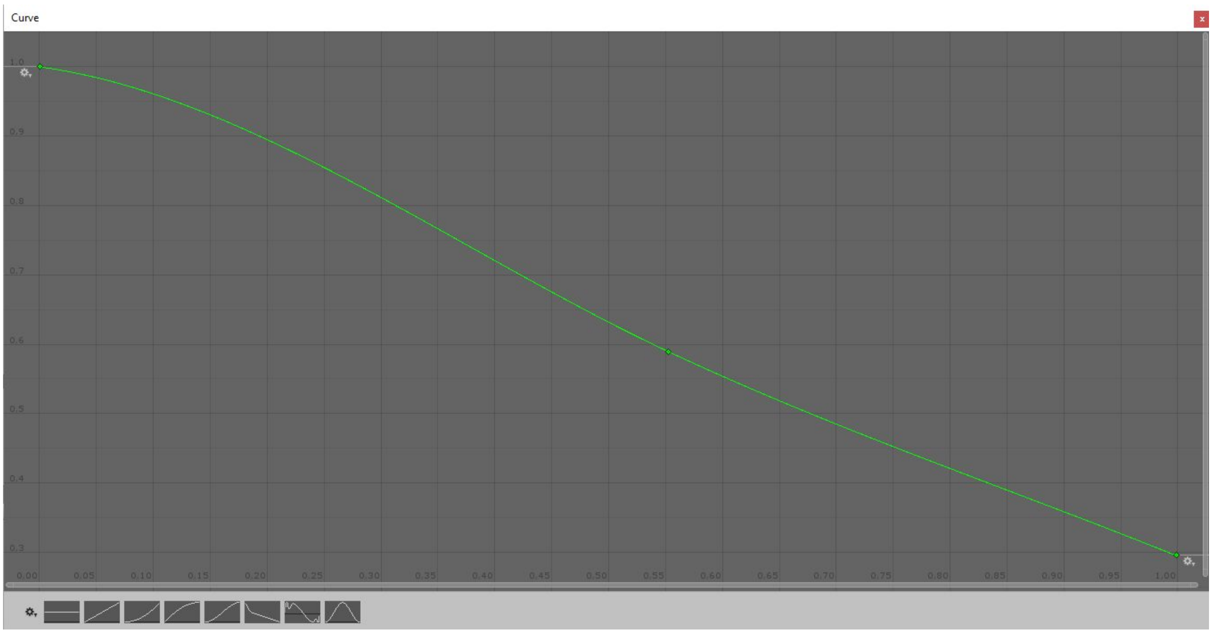
The maximum speed that a car can achieve.

Max Steering Angle

Max steering angle of all wheels that are connected to the axle with Steering boolean set to true.

Steering Curve

This curve determines how the Max Steering Angle is modified depending on the speed of the car.



On the horizontal axis is the speed of the car where 1 is the max speed of the car and 0 is 0km/h.
On the vertical axis is the steering angle where 0 is 0 angle and 1 is the max steering angle.

Center Of Mass

This sets the rigid body center of mass relative to the pivot point of the transform rigid body attached to.

Keys

W - forward

S - back/brake

A - left

D - right

Space - handbrake

E - interaction

L - Lights On/Off

Tank Controller

Set up

The first tab is Set up. Here you can set up tank gun, belts, and other essential variables.

Belts

Belts are controlled by bones that move dependent on wheel colliders.

Right Belt Renderer

This is Skinned Mesh Renderer, which is necessary to simulate belt movement with bones.

Right Belt Wheels

Here you set Wheel Colliders for the right belt, but only that is touching the ground. Plus for each, you select a bone that will have this wheel effect on.

Left Belt Renderer

This is Skinned Mesh Renderer, which is necessary to simulate belt movement with bones.

Left Belt Wheels

Here you set Wheel Colliders for the left belt, but only that is touching the ground. Plus for each, you select a bone that will have this wheel effect on.

Cosmetic Wheels

These are wheels that are not on the ground. It's here to make them rotate.

Armament

Tank Gun

GameObject that will be rotated as a mean to aim vertically.

Weapon

This is a Weapon Script which controls shooting.

User Interface

Like every vehicle, it has a Health UI foldout in which you can choose the type of health UI (Text, Slider, Text, and Slider).
Dependent on what you choose you will see those fields:

Text

This is TextMeshPro which shows health as text.

Slider

This is a Slider that shows health as a linear bar.

Crosshair Gun

This is an image, which is used for showing where a gun is aiming at.

Reloading UI

This is a ProgressBar script that shows reloading as a circle bar.

Speed Text

This is TextMeshPro which shows tank speed as text.

Interaction Setup

Canvas

This is the canvas that will be shown when driving a tank.

Player Slot

Position of the player while driving the tank.

Get Out Point

Position of player relative to the tank when you exit.

Blow Up FX

GameObject prefab that will be instantiated on tank destroy.

Virtual Camera

Virtual Camera that will be switched to when you enter the tank.

Stop Interacting Key

You can choose which Key you want to press to exit the tank.

Tank Turret

This is a GameObject that represents the turret of the tank, that will be rotated as means to aim horizontally.

Statistics

In the second tab are variables that change statistics and characteristics of the car.

Health

Amount of tank health points.

Motor Torque

This is torque in Newton-meters that will be applied on belts.

Brake Power

This is torque in Newton-meters that will be applied as brake power on all axes with Brake boolean set to true.

Max Speed

The maximum speed that a tank can achieve.

Turn Speed

This is how fast the tank will turn in degrees per second.

Turret Turn Speed

This is how fast the turret will turn in degrees per second.

Gun Turn Speed

This is how fast the gun will turn in degrees per second.

Elevation

This is the maximum angle a tank can raise its barrel (in degrees).

Depression

This is the maximum angle a tank can lower its barrel (in degrees).

Center Of Mass

This sets the Rigidbody center of mass relative to the pivot point of the transform Rigidbody attached to.

Keys

W - forward
S - back/brake
A - left

D - right

Space - handbrake

E - interaction

Right Mouse Button - stop turret and gun rotation

Left Mouse Button - Fire

Plane Controller

Set up

The first tab is Set up and here you can set up plane wheels, wings, guns, and other essential variables.

Wheels

Here you can determine the number of wheels in a size field. For each wheel, you must select the wheel collider (Add wheel collider component to empty GameObject and set the model of the wheel as the child of that GameObject to be visually rotating).
For each wheel setup these booleans:

Steering

Turn this on if you want this wheel to be able to turn.

Brake

Turn this on if you want this wheel to have brakes.

Wings

Wings are physics-based. You need to set up their position and area in square meters.