

Lighthouse Project

User Guide (Version 2.0)

Contact: ripunjoykalita02@gmail.com

Overview:

These Lighthouse and boat blueprints are specifically constructed to give the user a plug-and-play experience for their projects. This project consists of a primary model of Lighthouse with basic Lightning and Rain blueprints and also contains a boat master blueprint, 8 template blueprints, and has an audio setup for powered boats. The controls are right now using keyboard and mouse only.

Link to videos and tutorials:

Demo: <https://youtu.be/rWXnOQ1SwW4>

Demo2 Boats: <https://youtu.be/lvgoEzBuszc>

What each variable does: https://youtu.be/TzULKVKt_UM

Guide to implement custom models: <https://youtu.be/WSDf5bh-hI4>

Playable Demo: [Link](#)

Features

- One LightHouse model with base and Generator Room.
- Rotating Light Blueprint for LightHouse.
- ThunderStorm Blueprints with Niagara Rain particles.
- One master boat blueprint for easy tweaks.
- 8 boat templates for easy integration to projects.
- Inbuilt fully functional HUD with Direction, Power, and Speed Output.
- Properly commented blueprints for easy reading.
- Flexible control with 16 intuitive variables.
- 8 boat models included.
- Convert your boat to a submarine (fully functional) or hovercraft(experimental) with the change in a Boolean.
- 72 textures.
- 18 materials + 22 material instances.
- 2 basic water splash particle systems.
- 9 sounds included.

Contents:

Variable Definitions	2
Controls	6
Inbuilt Optimisation and components	6
Integrate your boat model	8

Variable Definitions:

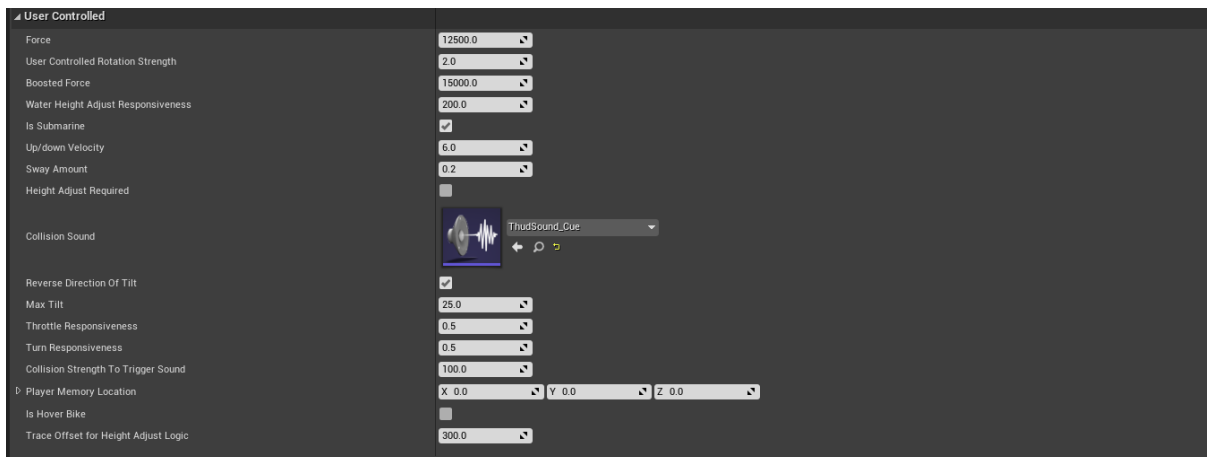
LightHouse: There are four variables that can be controlled.

1. Rotate: Turn on/off Rotation for LightHouse light.
2. Rotation speed: Keep it around 0.2-0.4.
3. Fogsheets: This turns on/off actual volumetric lights or the cheaper fog sheets.
 - a. When on: The fogsheets are visible and the fake light is visible even without any fog in your level. Recommended settings as its much more performant.
 - b. When off: Your fog sheet is hidden and the light turns on. Requires atmospheric fog or exponential height fog for it to work and the view distance is limited.
4. Scale: Self Explanatory

Thunder: There are a total of 5 variables and each of them are self explanatory.

Boats: There are a ton of variables to give you the maximum flexibility to tweak the physics of the boat to your liking. The boat runs on tick but don't worry about the performance as I will discuss that later here how it's optimized. So, let's dive deep into the variables.

There is a total of 16 variables in this project and each has its use cases and importance. There is a normal range for some of them which you would love to stick to keep more realistic. One thing to note is that the variables that you want to edit are termed under the user-controlled section and you should not normally touch any other variables.



1. Force: This variable is more of guesswork and the typical ranges are something like this:

Small Rowboat (<300 kg)	2000-3000
Rowboat big (<600 kg)	3000-4000
Powered boat (~1000 kg)	6000-9000
Submarine (~5000 kg)	15000-20000
Very large ship (~20 ton)	>35000

These are some reference values only and depending on your taste or game mechanics you may want to change them. The higher the force the higher is the final speed.

2. Boosted Force: If your boat needs to have a boost action then this is the variable you want. Set it to around 1000 to 5000 higher than your force value according to your requirement.
3. User-controlled rotation strength: It determines the max turning rate of a boat. If increased it will increase the strength of rotation about the z-axis.

Small Rowboat (<300 kg)	0.8
Rowboat big (<600 kg)	1
Powered boat (~1000 kg)	2
Submarine (~5000 kg)	2
Very large ship (~20 ton)	2

4. Throttle Responsiveness: It determines how fast you accelerate. Even if you give a ridiculously high number in your force and a very low number here then it won't accelerate. Rule of thumb: Force- Max speed, Responsiveness: Acceleration

Small Rowboat (<300 kg)	0.4
Rowboat big (<600 kg)	0.5
Powered boat (~1000 kg)	0.8
Submarine (~5000 kg)	0.2
Very large ship (~20 ton)	0.025

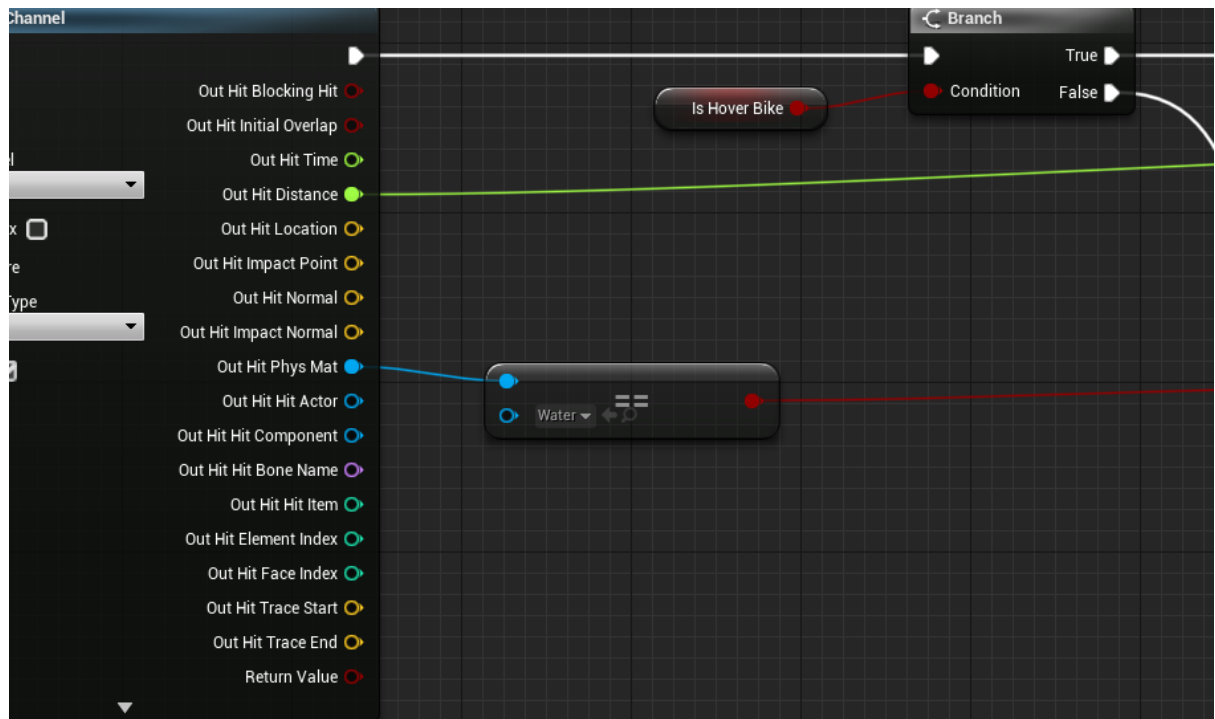
5. Turn Responsiveness: Same as throttle responsiveness but determines the speed of turning instead. How fast your boat will reach the desired rotation about z. See the variables explanation video seeing it in action.

Small Rowboat (<300 kg)	0.6
Rowboat big (<600 kg)	0.6
Powered boat (~1000 kg)	0.8
Submarine (~5000 kg)	0.5
Very large ship (~20 ton)	<0.1

6. Max tilt: It will define the tilt angle in degrees when the boat has reached max turning speed. Normally I would keep this around 20 but you can go higher for speedboats to give the Arcady feel.
7. Reverse the direction of tilt: Reverses the direction of the tilt. Suppose by default your powerboats will tilt towards the right when going right. But in big ships due to the momentum, it will move opposite to the direction you are going. So, you should then consider turning on this variable.
8. Collision strength: Defines the minimum collision strength to trigger sounds. It's experimental and the values are currently not that refined but you should get the desired effects by assigning values like 200 for small boats and 800 for very big boats. There is still some work remaining and if you all want me to reiterate on this then sure.
9. Sway amount: Boats randomly sway in the water as you know. So, a simplified implementation of the same.

Small Rowboat (<300 kg)	0.8
Rowboat big (<600 kg)	0.7
Powered boat (~1000 kg)	0.5
Submarine (~5000 kg)	0.2
Very large ship (~20 ton)	<0.2

10. Height Adjust Required: (Performance may have an impact) This does a line trace at a fairly high rate so keep this in mind. In my scene, it was working well without any visible performance loss. In CPU-bound games it might be heavy but haven't tried. It will work on any surface with the water physics material. To change the material for your project change it here in the macro- "Height Adjust".



11. Water Height Adjust responsiveness: Keep it between 250 to 400 for faster boats. For very slow boats you can go as low as 50. It just defines the rate at which the tracing is done.
12. Is submarine: Toggles whether it is a submarine or not. As simple as that 😊
13. Up/Down velocity: For submarines when you press W/E it defines how fast the submarine adjusts its height. Keep it below 2 always.
14. Is Hoverbike (Experimental only): Turn your model into a hoverbike and it will conform to any surface. It's the same as Water height adjust but minus the physics material limitation. But it's experimental and not suitable for rough terrain or high speeds.
15. Player location memory: Working on this. Right now, it spawns the player back to where he boarded the boat. Once implemented this will be useful for something like spawning back the player near to port or something similar.
16. Trace offset: How far ahead is the trace done for the surface. Hoverbike? – keep it higher around 450. A boat that adjusts to undulating water surface? - keep it lower and around 10.

Some combinations for the three Booleans:

Variables			Consequences
Height-adjust	Is submarine	Is Hoverbike	
False	False	False	None
True	False	False	Adjust to water or defined physics surface only
False	True	False	Submarine (Best setting)
False	False	True	Won't work. Just a normal boat.
True	True	False	Submarine (Unnecessary low performance)
True	False	True	Hoverbike (Experimental)
False	True	True	None
True	True	True	God knows :p

Controls:

WSAD – Movement Basic for both 3rd person and boat

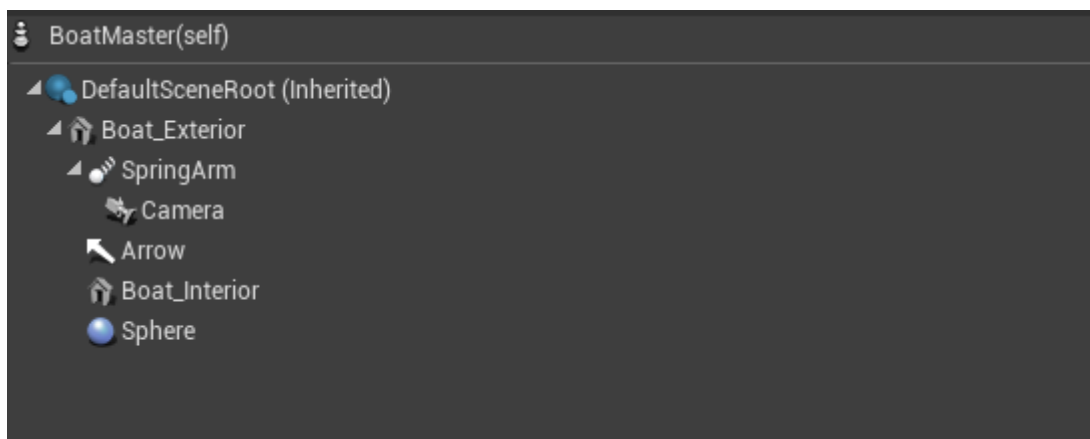
F – Enter/Exit boat – You must look at it and it does a line trace

Q/E – Submarine controls UP/Down

Left Shift – Boost

Inbuilt Optimisation and components:

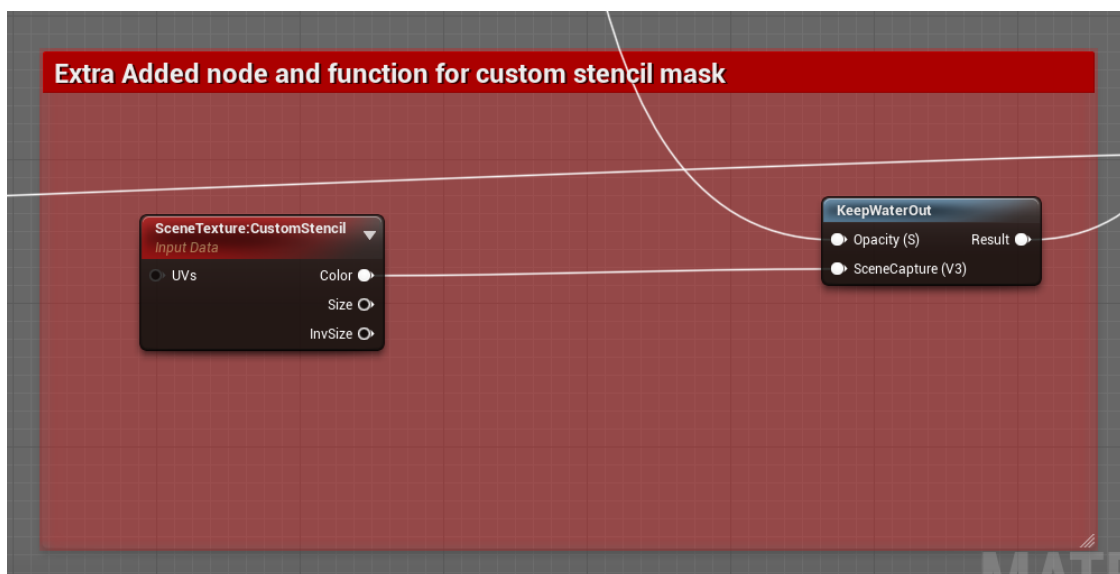
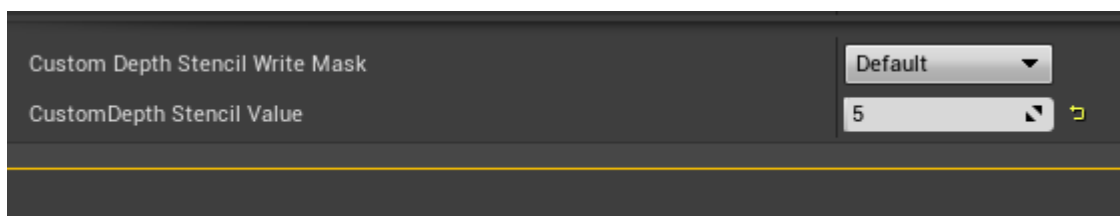
The boats work on tick events so you might think a lot of actors on the map will affect performance but the tick is only applicable when the player is around the boat. Let's check the basic structure of the parent BP.

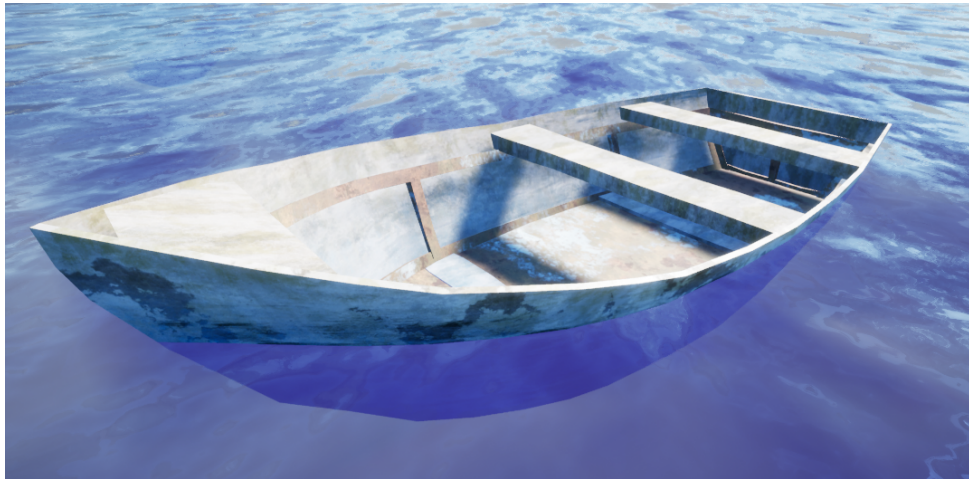
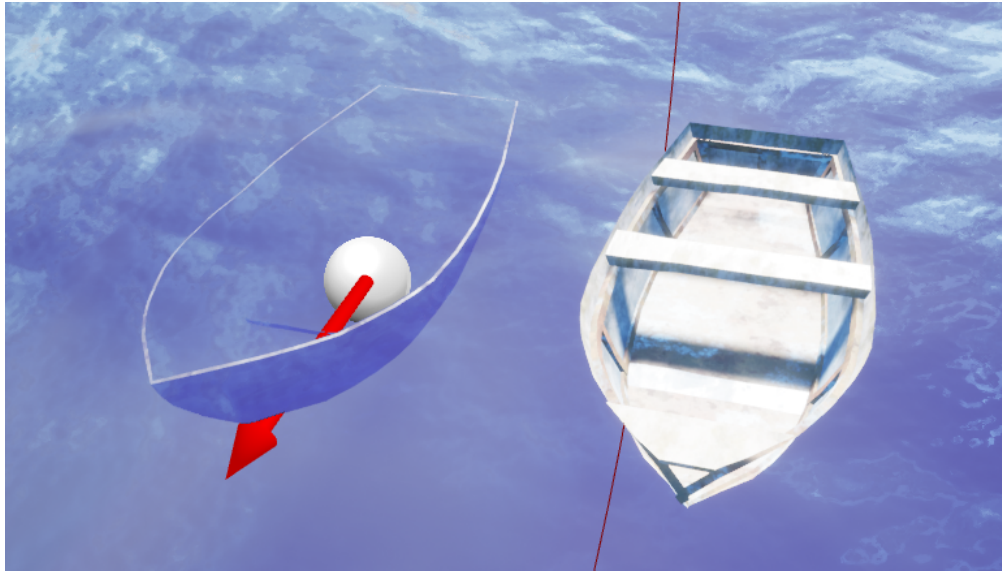


Boat Exterior: The main physics is connected to the Boat Exterior static mesh and to increase or decrease mass you need to use that static mesh.

Boat Interior: Now this is interesting. Suppose you have a small boat and its interior is visible then obviously the water will be rendered inside the boat also. So, how to not allow that. It's simple, just put the interior mesh inside this Boat interior Static mesh component.

How it works: It has a custom stencil value and therefore can be got from the captured scene post-process node. This then can be used with transparent materials to avoid the water being rendered over this mesh. The water material in this project is already set up. You can check how it's done, I am not a materials guy so cannot explain this exactly. I have defined a function for this and you can use this function in your water material if you have such boats. You can see the boat parts how they are divided up in the following pictures.





Spring arm and Camera: You all know, I am sure.

Arrow: Please keep this always pointing towards the front otherwise it won't work.

Sphere: (Optimisation) defines the region when you enter it, only then it will be ticking. This way you don't need a boat to be ticking across the other half of the map. It will stay still and won't simulate. You can use a capsule in place of this. I might change it if I feel it is necessary.

Integrate your model:

It's as simple as adding your boat to the Boat-Exterior component and clearing the Boat- interior component.

Notice: Please use a mesh having X as the forward axis and if it can't be changed, please turn of TILT

ANGLE to zero in the blueprint to avoid any weird animations. Currently working on this with priority.