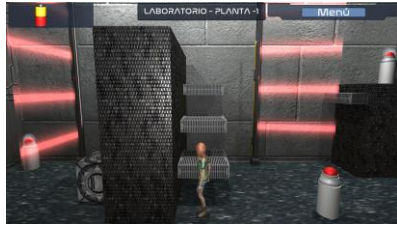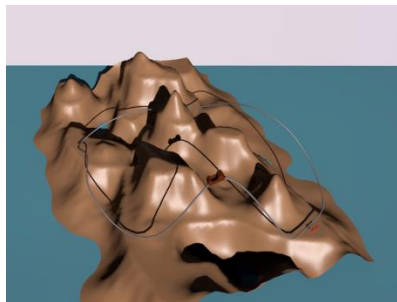# Clara Peñalva Carbonell – Breakdown List

I have developed all the projects during the years 2015, 2016 and 2017, being part of my university curriculum. You can see some of these projects in my [Demo Reel](https://youtu.be/QFcJSs_ceJ0). (https://youtu.be/QFcJSs_ceJ0)
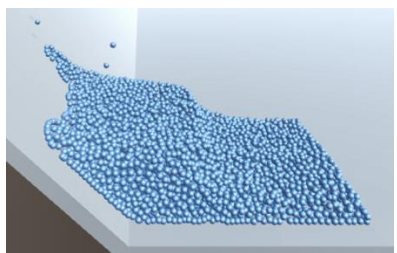
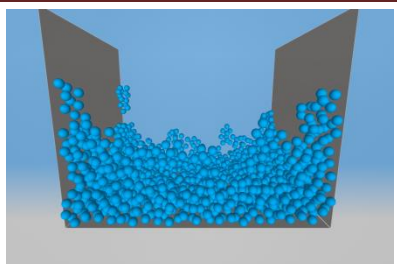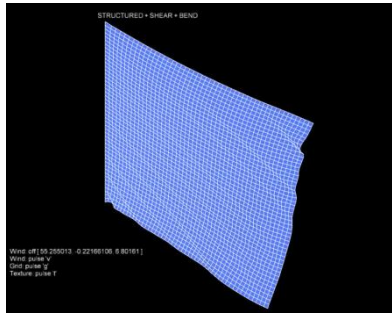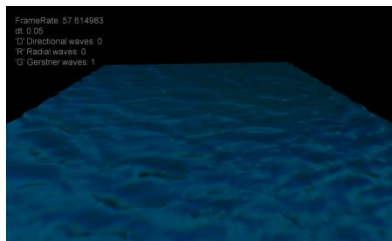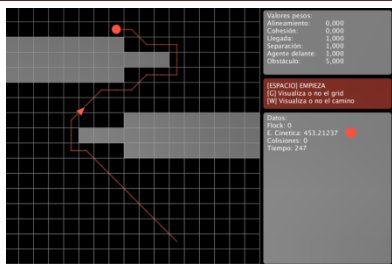|	
	As a final project of Video Game Development, our 5-person working group created a video game using **Unity** and **C#**. Specifically, I was responsible for creating the internal structure that handles all the data of the game, such as collected objects or areas already visited. As well as programming the interactions of the player with her virtual environment. The game was developed for PC.
	Here I developed a Plug-in for **Blender** in **Python** to describe the railway truck's path. This plug-in consider the positions of all "empty" elements in the scene, and calculate his ideal path and velocity of the railway truck, considering the mechanical force. This path is inserted into the corresponding frames. In addition, I used quaternions to orientate the railway truck and different types of interpolation, like Hermite Polynomials or Catmull-rom Splines.
	Employing a simple 3D model in Blender, I have introduced a skeleton with **Inverse Kinematic**. Moreover, I have use **MOCAP** technique to capture difficult movements, like this girl climbing. All dates have been processed with Python.
	Reading Technical Papers, I have implanted the Smoothed Particle Hydrodynamics (**SPH**) model, which simulate fluid mechanics. Under Processing and **Unity 3D (C#)**. I will work more on this project in the coming months because this takes part in my BSc thesis.
	I have developed the Position Based Fluids (**PBF**) model, which is considered an improvement of the SPH because it ensures the incompressibility of the fluid. Under Processing. (**Java**).
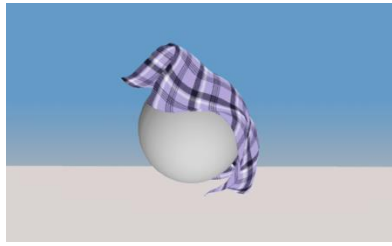
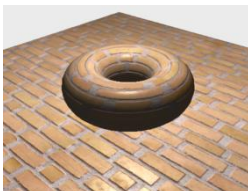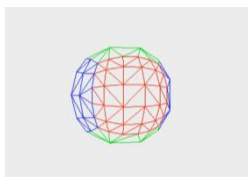|	
	I have implemented Position Based Dynamics (**PBD**), with the same procedure as the SPH, under Processing and **Unity 3D**. This model allows simulate soft objects like cloth.
	I have simulated **Soft Objects** based on Mass-Spring Lagrangian model with Processing and Java. In this example you can see a flag waving in the wind.
	In this project I have implemented **ocean waves**, like the animated behavior of Gerstner waves, directional and radial waves. Using a mesh and modifying the surface. I have used Processing and **Java** to develop it.
	This project contains two parts: **Steering Behaviors**, where you can see multiples agents flocking together. And the second part, where it's implemented the **A*** algorithm and it's introduced into the steering agents, having a **Pathfinding** simulation that can avoid obstacles. I have used Processing and Java to develop it.
	I have done several projects where **GPU** is used. In this section I will show different little projects, where I've programmed various types of shaders with **C++** and **OpenGL**. - Changing the level of tessellation depending on the distance between the camera and the sphere. - Exploiting the use of fragment shader to implement the gloss map and bump map techniques. This adds realism to the scene without increasing the geometric complexity.