CATEGORIES:

PLAYER CONTROLLER

- Added option in the current feature to make the player to always look in camera direction to disable this state when the player is running or sprinting, similar to resident evil 2 remake. LIke that, when the player walks, he uses the strafe movement and when he is in run or sprint state, he can rotate toward the direction of the input respect the direction of the camera, like if the option to look toward camera direction is not active in this state
- Added the elements and options to use an injured state in the animator, making it very flexible and being able to use it in different situations. I have added a random idle component to allow to configure a list of random idles, its duration and a random option to play them in any order. The player can also use weapons and aim in this injured mode as well
- Added improvements to the animator system to make a better transition from free moving to strafe mode. This has been applied in the second version of the animator and now that will be the default and main animator of the asset, so for unity 2018 and higher, there won't be a need to change the animator of any character, since it will be already using that
- Improved head track management, now it is smoothly enabled/disabled according to the situation and also, the use of menus doesn’t deactivate the head track of the player like happened previously
- Improved head track with an option to configure the max distance from the head to the transform used to follow within the camera direction, so if this transform is far enough, the head track will ignore it (for example, if the camera is a far away state from its regular position, the head track will detect it as a higher distance than the allowed one, avoiding to look in a awkward direction)
- Added a roll action in the action system, it can be used in any direction, taking the input as the roll direction. The action system can also change the player capsule scale, so for example, the player can pass below obstacles
- Added improvements in the movement of the player, by adding more customization options for the input on the lerp speed of vertical (W S) and horizontal values (A D) (or course, taking into account gamepad and touch controls) for regular movement and strafe in both free view and locked view, including tank controls. So this allows to make the start and finish of the movement in the player faster or slower according to what you prefer
- Added option to configure an amount of time where an object can’t receive damage (invincible) in the health component. This is used for example in the roll action, so when it starts, this temporal invincible state is called with a certain amount of damage as parameter in the function, combined with the delay list of events in the action system, allows to use any delay amount for this state to be activated

PLAYER CAMERA

- Added an option in the health to avoid showing the health bar above objects by default and include events on the lock on target system to configure remove events to trigger (which is used for objects which are not connected between each other, so instead of using events, it uses the remote event system)
- So one of those remove events can be configured to show and hide these health bars, activating the bar on the current target locked by the player, similar to games like dark souls, instead of showing all the health bars of at the enemies visible on screen at the same time
- Added option to rotate the camera x amount of degrees in the Y axis with input. This is used for example in resident evil games, to make the camera to rotate towards the opposite direction, usually 180 degrees with respect to the player. In this case, instead of 180, any amount can be configured to rotate each time and as the main direction to use, the forward direction of the camera or the forward direction of the player plus the amount of degrees
- Added option to use the surface detection system to set transparent the surfaces found between camera and player that was used previously in locked camera to use it on free camera as well. So objects with collider can use a new layer to being detected by this system and avoid the camera collision detection to place the camera outside of the view of that object, having all the other collisions totally active and working as any other type of regular collider
- Added option to start the game with a camera effect already active, like pixelated, or black and white, etc…
- Improved check surface detection system to set them as transparent for locked or free view, with a better management of surfaces detected and the change from original to transparent (or other shader) and vice versa
- Added option to configure a default camera reticle in case there are no more reticles active at that moment (from elements like weapons, powers, etc…). It has options to use it separately only on third person, or first person or both
- Added check to the transparent surface system to turn the surfaces found to their regular material when the player is using a device and turn them again to transparent once the player stops to use that device. This avoids that any possible surface keeps transparent when the player is using a device, so surfaces can be seen normally on this mode
- Added a culling system for the player, so when the camera is close enough to him (all configurable in the editor), the shader of the player's meshes changes and is configured according to the remaining distance to fade it smoothly and see through him

LOCKED CAMERA

- Added a new shader for the detection of surface between player and the camera for the locked view and you can see here the result (the previous transparent option can be used instead of this one if you prefer it)
- Added option to change the fixed camera position closer or farther from its original position according to the distance of the player with respect to an empty transform on the scene. This allows you to make camera pans easily, like many top down/isometric/2.5d games do, according to if the player is moving closer or farther to a specific object or position. It includes options for the movement speed of the camera, distance multiplier, clamp values (forward and backward) and if this movement is directly or indirectly proportional to the distance of the reference position/object and the player

WEAPONS

- Added integration with this slice system (https://github.com/DavidArayan/ezy-slice), making also a plasma cutter weapon, using the behavior system for both weapon and projectiles, allowing for example to rotate the weapon rail, so the cuts can be made in any angle
- For this, just the slice system was imported to the project where I work with GKC and made a new projectile type behavior. Nothing extra was needed, besides adding a weapon dedicated to that type of projectile (taking inspiration from dead space and metal gear for a rotating cannon weapon), allowing to make slices in any angle and having total control of the rotation
- Improved reticle management, adding options to configure if the reticle is used separately on first and third person and when aiming on each view along with a main option to enable or disable the reticle it self from all weapons, so it can be used on any of them or none
- Added the slice system to any object which can apply the slice on collision with other objects, similar to the saws of half life 2
- Added remote event options on player weapons and vehicle weapons to trigger events for the damaged objects by those projectiles
- Added main options in the player weapons manager to ignore the collision detection in all weapons, with separated options for third and first person, so in that case, even if the collision detection on a weapon is active, if the main ignore option is enabled, that weapon won't use the collision detection
- Improved spread system and options, making it more easier to use and more customizable, including spread option for third person when the player doesn't aim on this view
- Improved movements for actions like draw and keep weapons in first person and activate the edit attachments for weapons in game, with smoother movements for weapons and camera, to make them more natural and better

- Added option to use the smartphone tool in third person more easily, through a set of options that allows to change the position, scale and location of a canvas, in this case, the smartphone screen from its 3d world position to a fixed position in the player's HUD (totally configurable to set the position and scale of that screen on the HUD). This makes it easier to use the smartphone in third person, similar to games like GTA V. Also, the weapon system allows now to unlock the cursor of the game in third person as well (previously it was only possible on first person) in order to use the UI buttons of the smartphone, to move between its apps. This can be also used to press button and elements on any weapon/tool and can be customized separately for each view and for each weapon
- Improved aspect and organization of the player weapon system, in order to improve how the settings are shown and order them by category, like projectile settings, or events options or particles and sounds, ammo, etc... allowing to show all the settings together, like previously or just by these categories

INVENTORY

- Added event options on the inventory bank system to trigger events if the inventory is empty. This can be used to trigger events when picking inventory objects from dead enemies from their "pockets"
- Added new inventory object type to affect to any stat of the player (to increase or decrease its value, like for the current health for example)
- Added grenades as a new inventory object as a regular object with a health component attached, so if when the object is a pickup in the scene and receives damage, it will "die". This death triggers events to use the external shake camera system and the destroyable object component which creates an explosion applying damage and physics forces around a radius, so all combined creates a grenade pickup which can be stored in the inventory and being damage and explode too
- Added component which allows to check if the player has units available of any inventory object, searching by name and triggering events in case the player has units available and in the other case. Along with that, the component allows to remove any amount of that object and trigger events in the case of successfully removing that amount of units from an inventory object and also in case there are not enough units. This is used for example for the new grenade inventory combined with the ability system to throw grenades. Like that, the ability first checks if there are grenades available in the inventory and in case positive, the ability to throw grenades can be activated. Once that is done, one unit of the grenade inventory object is removed from the player's inventory
- Added option on the pickup objects to ignore the option on the inventory manager to pre visualize the objects that are going to be picked before storing them in the inventory (if this option is active). This ignore option can be configured separately in each pickup object, for example, to use it only on certain objects, like the quest items

- Improved category selection in the inventory menu, allowing to see the objects filtered by categories, with a better code for that component and with a couple of issues fixed related to the option to show multiple categories at the same time. Also, the scroll bar is not reset properly when a new category is selected or when the regular inventory menu panel is enabled again, moving back the inventory objects grid to the top position again
- Added options in the main inventory list manager to save the current list of inventory objects configured in a scriptable object. This allows to save this info into an external file outside the main inventory manager component itself, making it easier to move this info from a previous version of GKC to a new one when you update the asset in your project or want to move this info from a prefab to another. This also allows to load the info from any of this file type (scriptable object) to the main inventory manager, to have the full save/load functions and use them as need, allowing to use as many files of this inventory info as needed
- Improved combine option on inventory, allowing to combine weapons and ammo without need to have that weapon equipped in that moment

VEHICLES

- Added options to fade vehicle parts or not
- Improved management of vehicle pieces when a vehicle is destroyed and broken in pieces, having a better performance, making the code simpler and faster and removing some unnecessary checks on that part (once a vehicle is destroyed).
- It also includes options to configure certain vehicle parts as the main "pieces" to be used for the destroyed state, similar to games like GTA and others where the main chassis and the rest of elements like doors, wheels and other parts are separated from each other (another option also allows to don't remove or fade these vehicle pieces if your prefer that instead of removing or fading these elements). Also, the collisions between passengers inside the vehicle, their ragdolls and the vehicle part colliders is managed better now, toggling between ignoring collision to true or false according to the situation. The vehicle parts to drop option is very customizable, and allows to configure any amount of parts and set them as needed
- Added option to simulate collision to test the launch passengers on collision option in the vehicle hud manager component
- Added vehicle creator wizard, similar to the character creator, to create new vehicles using the current prefabs configured, allowing to replace the meshes of these vehicles for new ones in just a couple of clicks

PICKUPS

- Added new pickup object type to affect to any stat of the player (to increase or decrease its value, like for the current health for example)
- Improved pickup icon manager, allowing more customization now, like setting custom icon prefabs or custom icons textures for each pickup separately, even if they are in the same category of pickup

INTERACTION ELEMENTS

- Added option to configure a list of custom interaction icon panels (the one showing that an interaction is available, showing the name of the object, the action to do and the interaction key), so instead of using always the same, each object to interact can have its own customized interaction panel, making simpler or more complex versions of these panels
- Added option to configure a maximum distance and a maximum angle between the object to interact and the player, so if the is not close enough or his forward direction is not close enough to the position of the object, he won't be able to interact and neither the interaction panel info will appear on screen. This can be configured separately on each object
- Improved detection of objects to interact, for objects that pause player actions, like using an inventory bank, a code terminal and any other device that usually changes player camera position or pause his actions, avoiding to add new devices to the detected list while the player is already using another one. This is important so the system doesn't take a new device as the current to use instead of the previous like. An example of this is the next: if you're using a hiding spot while there's an NPC to talk to standing next to the hiding spot you can accidentally trigger dialog instead of hiding. This can also prevent you from leaving your hiding spot. This is now solved with this improvement
- Added option to only allow to interact with an object if it is visible on the camera, so that combined with the current options to use min distance to interact with an object and the max distance of the object from the camera center allows to work in the way you need, so the player can't use objects that are behind of him. This is a general option so a separated option will be added on these devices, so maybe for specific objects you want to use the other way currently configured
- Added options to use a fixed lerp speed on the move camera to device and move device to camera to make these movements smoothly according to the distance to the fixed camera position or fixed device position or with a fixed lerp speed (smoothly as well), with different speeds for third and first person view
- Added a component that can be attached to any object and which can be used by a system that can be attached to weapons to detect the first component attached to objects through a raycast using the main camera (or any other transform) and which can activate events through it. So for example, events can be configured in any object and

triggered by a checking of a raycast from the main camera, for things an ability used to trigger these events. For example, break crates, unlock a door, etc… just by pointing at them and using the ability/power/tool/weapon configured for it

AI

- Improved checking of the current angle between the position of the target and the forward direction of the AI, in order to only activate the current attack when that angle is inside a range. This can be disabled to attack without checking this angle between target and AI
- Improved noise detection and patrol system for the AI. When it detects a noise while moving through a patrol, the AI will check the noise position and be there for a certain amount of time. If no target is found (configurable in inspector), it will resume the patrol, going to the closest patrol point and keep on the regular patrol state
- Optimization improvements on AI related to the management of targets to attack and how that info is obtained, reducing the use of resources and making the code more light, simple and efficient
- Added option to allow to check the distance with the partner (the player) in case it is located and a place position target is assigned. So similar to games where the player is going to a location with a companion AI and the AI is leading the way, waiting for him if the distance to that player is higher than X
- Added option to configure if the AI checks the decibels amount of the noise that it detects (the noise settings for the object which produces it has an option now to configure the decibels of the noise, with a range from 0 to 1). So each AI can have a different amount of decibels which is able to detect, like one which has a very good hear, with a value of 0.1, so it will hear any object equal or above that amount of decibels or other with 0.9, so it will be only able to detect loud sounds, like firing weapons, but not lower ones, like a foot step
- Improved view trigger on the AI, configuring a delay time to activate again that view trigger (the capsule used for check suspects with a delay) once the AI has lost track of all the previous targets to attack, so the AI doesn't check for targets immediately in the same next frame after the previous event. In this state, the sphere trigger used as range for detection targets stills active as usual
- Added an option to AI to check if the player is far away enough to stop moving toward a target position where that AI is leading the way and waiting for the player to be close enough to keep moving. Like that, the AI will change its target position toward the target and once it is close enough to him, it will resume its previous target position, while it keeps checking if the player is close enough to keep moving towards the final target position
- Improved functions and code for the AI type which run away from its current enemies instead of fight or attack them, moving properly in the opposite direction and checking

the proper position to reach checking if it can be navigated properly through the navmesh system
- Added prefabs for the AI which uses powers to attack its target, instead of weapons or combat and that AI type can be selected in the character creator as well

MISSION SYSTEM

- Added option to play custom sounds on every sub objective complete in a mission
- Improved mission management for the options to use objective icons on screen to show positions or targets where the player must go, used in many games where the player is able to see an icon following an object or position on screen, so he knows the location to reachm, allowing to show these positions or objects position one by one, according to the current sub objective on the current mission
- Added checks on the code to avoid to start a mission if the player has been assigned manually properly (in those case where a mission is not activate by a trigger but by another element, like an event, so the player is assigned manually in the mission system component directly)

INPUT MANAGER

- Improved touch controls for the joysticks, including a better system to check if the touch zone of each joystick is being pressed, using now an actual recttransform to configure this touch zone, making easier and faster this configuration and adjusting properly to any resolution, since it is managed directly by the UI system
- Improved management of the cursor limit on screen when a gamepad is being used and the menu pause or any other ingame menu is used and activates the movement of the PC cursor through the mouse. Now the limits of the screen size is calculated properly, no matter the resolution of the screen and in both, editor and build. The cursor position is also reset properly when any of these menus is opened, setting the cursor in the center of the screen
- The code part that manages the action panels has been moved to a separated script (before it was in player input manager) and I have added action panels to the player modes (those configured in the player states manager) to show basic/main input actions, for the modes of weapons, powers, combat and simple mode, so you can have an action panel active by default to show main input for the current actions available
- Of course that aspect can be totally customized, to put each action panel on any location of the screen and with any style/size
- Improved input for the touch system to change powers, player's weapons, vehicle weapons, select or edit powers on the powers wheel and the weapons wheel along with the swipe action, having a more responsive control and a better management through the code, including to work better on any resolution

- Touch controls are properly disabled now on elements like cutscenes or dialogs
- Added option on the mouse cursor controller (to control mouse cursor with gamepad) to use left or right joystick and a function to change this setting at any moment ingame through events or any other function call

OTHERS

- Improved input management related to situations where the player actions are paused, like in photo mode, a cutscene, etc. avoiding from any action to be executed properly, with a general check in the player input system
- Improved menu management and checks and setting on the current state of the player menus during game
- Added option on the capture gallery menu to remove the current expanded capture which is selected
- Added simple player mode, being a new mode in the list of powers/weapons/combat, where these 3 modes are totally disabled and the player haven't any of those elements active
- Added debug option on health to heal any object by adding extra health amount at any moment with a button on the editor and an amount field to add that value
- Improved component inspector aspect with a better organization in each inspector and with fields separated by categories (settings, components, debug, gizmo, etc…)
- Added damage in area component which allows to detect and apply a certain amount of damage once or with a certain rate
- Removed some references of some components from some scripts, making everything more independent and light
- Added the check in the input to take into account if the movement or camera orientation is reversed in horizontal or vertical direction, so in menus, the cursor movement is straight instead of taking the orientation of the current input selected
- Improved grab objects system and gravity gun, making both components with less references to some scripts that were previously checked when grabbing an object, making them more independent
- Added option to use a general damage multiplier value in the health component, which can be used to decrease or increase the total damage received
- Added canvas group component, which allows to change the alpha value of a group of UI elements at once, for menus and other panels on screen. Instead of enabling or disabling a panel at once, it can use a smooth fade in and out instead. Of course, this can be disabled if you prefer, to just keep the same enable/disable at once each UI panel
- Added option to use random delays between events (using a range of float values) to trigger each even in the event trigger system component
- Improved grab objects ability management, including all others components that allows to grab and carry objects around, like the gravity gun example

- Added prefab to activate a noise (that can be detected by the AI) that can be called by any element, like an event, the event trigger system, a function call from another component, etc… Examples of this is the radio system to examine, which has configured the play button to call this noise system or inventory places where once an inventory object is used, it can call to this noise system, like an engine that has been turned on and it is making noise which will alarm to the enemies
- Added option to use hover check over point and click elements on that view, to activate events on hover enter and exit. An example of this is activate and deactivate a head track target on that point and click element, so when the mouse is over that object, the head track of the player can look to that direction if it is inside his range of view (angle, direction and position)
- Added option to use a list of tags to ignore for the close combat system, so in each character, you can customize if certain tags are ignored to receive damage, for example, the enemy AI unable to damage other enemy AI, using the tag “enemy”. Same can be done with friend AI, to avoid “friendly damage”
- Added option to use a limit amount on the vending machines to spawn objects
- Added an option to the radio system to use internal files for the song list, so it can use files from inside the project and not only external files. This internal file list can use .mp3 files (external files can be only used with .wav extension), since these are located inside the project and of course, the build of the game
- Improved radio system, which now allows to configure an extra list of songs which are located inside the project, so in that case, they can be in .mp3 format. So besides external files, you can use internal files for songs as well
- Added option to auto climb ledges in first and third person when the player is grabbed to a ledge, so he climbs it automatically without need to press the movement input to make that action
- Added option to disable the gravity arrow in the back of the player when the gravity ability is not active or being used
- Added option to configure critical damage on the list of weak spots that can be customized on the health component and which can be configured on any object. This critical damage uses a probability that can be adjusted between two values: 0 and 100. A random number with that range is generated, if the value is inside that range (the random value can be 35 and the range 20 to 40, so the damage would be critical). This option allows to kill the target if the critical damage is the result obtained, apply a damage multiplier or remove the total amount of health from that weak spot (each weak spot can use a certain amount of health and if that amount reaches 0, events can be activated. That this amount reaches 0 doesn’t mean than the main health amount is 0 as well, but the damage applied to that zone is removed from the main health value). This also allows the text of the classic “CRITICAL!” to be shown through the damage on screen component, which is used by the health system to show on screen the damage applied to objects, similar to games like borderlands
- Added options on the change player state system (component used to activate multitude of different elements and set different states and functions ingame through triggers and

events) to use an option to enable or disable the transparent surface detection system in order to fade surfaces between player and the camera for those objects configured with the layer “Transparent Mesh”

- Added option to configure and activate the remote event system of the player to the above component of change player state system
- Added options to use the remote event system when grabbing and dropping an object on the grab object and gravity gun components. So with this, you can call events remotely on that object without having events with some dependency of the player or other object external to the grabbed object. For example, similar to half life 2, the action to grab an enemy with the gravity gun and turn it into a ragdoll is made through that remote event system, with nothing hard coded for it. You can even smash them just by hitting them towards a surface, since the ragdolls are configured by default to receive damage from collisions
- Added the external elements of the action system in the main options bar on the top of unity window, to add new actions and triggers easily
- Improved a previous system which was already present on the asset (player options editor system), which allows to configure any setting ingame and in any menu, and with different types of fields, scrollbars, sliders, toggle, etc.... including to show text info, like amounts (for example, game volume or mouse sensitivity actual value through text). Also, any amount of fields to manage can be configured and another new element is that serialization has been added to this component, so the values configured ingame are saved/loaded between plays. Examples of this are the previously said game volume, mouse sensitivity, show the ingame actions and its keys, all the different touch controls settings (joystick sensitivity, use them as touchpad, if the joysticks snap to finger position, etc...), and you can add any other field needed, like video settings. Here an example of how it works and using it when saving/loading a game
- Added flashlight attached to player’s body which can be used directly by input (menu key at the left side of right ctrl) and which follows player’s camera position and rotation, with an offset distance, placed close to the left side of the player’s chest, working on a similar way to resident evil 2 and 3 remake

ISSUES/BUGS FIXED

- Fixed bug on the head bob system, in the part of the camera shake, which was disabled properly when the player used a device or anything that moves the camera from its regular position, so sometimes, if a shake was active on the camera, that shake continued forever once the camera returns to its regular position until some external element produces another camera shake, stopping that shake loop from the camera and setting a new one until that new shake is over
- Explosive barrels and crates didn't have configured the explode and break functions configured in the death event option of the health component that they use to detect and apply damage to them, like projectiles, collisions, etc….
- AI turrets didn't have the death events configured neither, due to the health component was improved to avoid the use of any send message function and only use events instead, but I forgot to configure the death in some elements, like the turrets
- Fixed some minor issues between some transitions between free view to locked view and vice versa in specific situations
- Fixed some minor issues on the death of the player in first person, which wasn't activating the input after getting up properly
- Flashlight ammo name is configured properly now, so the flashlight batteries can recharge the flashlight when it runs out of battery and there are flashlight batteries in the inventory
- Fixed issue when using devices, there are certain actions that can be performed when the player has the using device state as true (like when using the computer with password), like change the player mode (weapons, close combat, powers, etc…), open the main player menu (that option to open a certain menu from the list configured and change between them easily, like console games used to do) and others. That doesn't happen anymore
- Fixed issue on gamepad when opening menus, that were taking the first interactable UI element as the current object to use, so if you were using the A button from gamepad to press buttons on the pause menu or the ingame menus, there could be other behaviour happening. This fixes that issue, so the pressing buttons action on menus with gamepad works properly now, for any menu, pause or ingame
- The confirm take pickup action used when the objects to pick for the inventory is previously examined has configured the same gamepad button as interaction action button, allowing to pick objects for the inventory properly by default when using a gamepad
- Lockers and closets which have objects inside now have an option to spawn objects when they are opened, instead of only activating objects previously placed inside. This fixes an issue where old versions of weapons were placed inside of these lockers and closets, so now, these weapons are spawned with their current version of their prefabs. This of course is just an example, so instead of weapons, any object can be placed inside. Same has been done with the puzzle system examples, which had old versions of pickups placed inside, so objects on those positions are spawned properly now

- Fixed issue on ladders which in some situations, it wasn't detecting the bottom part of the ladder when the player was trying to exit from the ladder on that position, avoiding him to exit from the ladder through the bottom part. This is fixed now, allowing to exit from the ladder through the bottom part without problem
- Fixed issue on the ladder system, avoiding a previous bug that was playing the ladder footstep sounds on the air before touching the ground and when not really moving on the ladder (for example, when the horizontal movement is locked on the ladder, keeping the player in the center of the ladder)
- Fixed some issues on the AI related to the detection and management of vehicles as targets to attack when the player is driving, getting on or getting off from vehicles, so this is made properly now. Also, AI will resume its activity if the vehicle or target is dead/destroyed or out of range, like returning to its patrol state
- Fixed issue on inventory when combining objects, when the combined object wasn't added to the player's inventory

CHANGE LEVEL MANAGER INGAME

- Added option on the scene manager to change from one scene to another directly in the editor, with just a button click, without need to search the scene file in the project folder, using the level manager component. It just needs a path setting (where scenes are stored) and the name of the scene, including a check in case the scene name or path configured doesn't actually exists

DIALOG SYSTEM

- Added options to stop a dialog which doesn't pause the player actions in case he is too far away from the NPC which is talking. There are also options to set the next dialog group in that character in that case or play again the last one and keep from there
- The external dialog component allows now to only activate the dialog checking the ID of the character detected
- Added option to play dialogs automatically when the payer enters in the trigger if case the dialog is configured as a free type (those which doesn't pause the actions of the player of put the camera in a fixed position)
- Fixed issue on the dialog system in the option to use the interaction button to change to the next dialog line, which was not stopping properly the coroutine used to show the line text letter by letter/word by word
- Added option in the main GKC menu (top options bar of unity) to add a dialog content system to any character or object automatically, adjusting the couple of manual settings that need to be applied directly, so the dialog is ready to be used (adding the dialog text lines needed for that character) with just using that main option bar

PHOTO MODE

- Improved photo mode, including the use of moving the camera positions even on states where the player input is overridden, like in point and click. Also, the photo mode is now properly managed if it is activated when the player is driving a vehicle
- Added option to configure extra cameras to apply the same effect as in the main one, so if a secondary camera to render an specific element of the scene is active, it will have the same effect applied

MENU MANAGEMENT

- Added a new system which allows to configure different menu schemes for the inventory menu and change between them in real time by sending the name of the scheme to use (though it could be used for other menus). It has options to enable/disable any element and set a new position,size delta and scale as well, so it is very customizable
- For example, I have made a similar menu scheme to the inventory of Resident Evil 2 remake to use it when the player is on a place to use inventory objects (a new option has been added to open the inventory menu and set a different menu scheme, if this is not used, the other inventory places will be like previously)
- The regular inventory menu is the same as before and when the player interacts with the inventory place, the camera changes of place (using the previous move camera to place component) and enables the inventory menu to use objects as needed and it changes the inventory menu scheme to a different one

ACTION SYSTEM

- Added action system to allow to add and configure new custom actions triggered by places of the level, specific situations, input or all combined
- There are also other options like an event list with different delays, so you can trigger a group of events, each one with a certain delay, which can be accumulative (wait 3 seconds, then 2, then 4,....) or each value is checked with a main timer (like previous values but instead of 9 seconds, it would be 4, as the other events happens inside that range)
- The events can be used for stuff like play animations in elements on the level, like press a button, activate the change player states for things like make the camera look toward a certain direction, etc…
- Some new options allow the player to move to a certain position at the beginning or at the end of the action by itself. It will also include an option to after reaching a certain position, to rotate toward a direction, in case it is needed that the player faces a certain direction at the beginning or at the end of the action

- It has been very improved with options to before play animation 1) make the player rotate toward a certain direction (so that avoids any limit to make the player to face a direction previously to interact with an object) and 2) adjust better the player position to the place of the animation. Also 3) an option allows to select a bone of the player to place an object inside, like for example, picking an object from the ground
- it can be configured that the player obtains an inventory object, like a key, so both the action and the event that triggers are related to what happens in the scene
- Added the first example of the action system being used at any moment and any place with the player making a rotation of 180 degrees, similar to resident evil old games. So the system already allows to trigger these actions at any moment by input for example, so you can add new actions based on input without need of code too
- Added the options to use a stealth kill (which in reality, is just an option to play animations for two characters that are meant to be synced), so it can be used for more interactions between two characters. And also, I added an option to use the movement input during actions, to use things like walking on a cliff
- Added simple component to trigger custom actions in the player by trigger, with options to set if it is triggered on enter or on exit
- Added new component, player state system, which allows to configure different states based on animations and other internal settings, like injured, drunk, cold, hot, hungry, etc...to change the current movement style of the player
- Other options included to configure a camera noise in the camera, so it applies a random input, configure a duration for this state and the ability to configure the call to this state through the use of inventory objects, like a beer, which for example can refill health but cause this drunk state during a certain amount of time
- Added vault and running slide to the actions available, being made automatically when detecting the triggers to jump above an obstacle or below it without any further input rather that just moving toward those triggers
- Added options to keep weapons while an action is being played, including options to draw the weapon again if the player was carrying it previously
- Added an option in the action system for those actions which are made while the player can still move, like the roll for example, where the camera direction is ignored, so he makes movements based on his direction rather than the camera
- Added option which allows to set a list of actions that can be used randomly when the player activates a custom action like the roll, so instead of make the same animation, it can perform a different animation for the same action randomly, where he sometimes roll and other he slides, they are just randomly picked of a list of two actions for this example, and more could be added
- Configured an example of the event settings of the roll and the slide actions to activate the push object components, similar to the slide in first person, where objects and other characters can be pushed when doing this action
- Improved the dash/dodge/roll/evade system (it can be called in different ways according to the type of animations used on it haha). Now, it can be used on strafe mode, checking the current input movement direction and activating the roll in that direction. This check

can be done using only 4 directions to check (forward, backward, left and right) or 8, including diagonal directions, so the roll/dodge/evade can be done in all those directions if needed
- Also, there is a new option to change the current camera state while the action is active, allowing to set the previous camera state at the end or a new one. The roll can be also used while using weapons and while aiming with them, and in that state, the IK on the weapon will be toggled smoothly to disabled while the action is active to be activated again (in case the IK was active through the weapons or other state)

ABILITY SYSTEM

- Added an ability system to allow to configure any list of abilities for the player, and the first two that I have made is a cloak/stealth system, to being 100% invisible to enemies and move the shield system to being a totally independent component. This system is made using a main class with virtual functions and each ability inherits from it, so that allows to override easily the main functions and create new abilities very easily, including options to trigger events
- This can be also combined with the action system to use animations with these abilities, so this will be part of the magic cast system. Like other similar systems, it will allow to unlock any of the configured abilities by any means, trigger, pickups, events, etc... and keep that info between games, including the ability to upgrade them as well
- So basically this system allows to assign a current ability and activate it with a button key just for these abilities and select the current ability in a typical wheel menu. Also, these abilities will be able to be configured as active (like in the videos) or passive
- Added two new abilities added, the teleport (which was previously part of the other powers component, but now it is in an independent script) and the gravity control can now be used as another ability
- Improved teleport, setting the on ground state to false while the teleport is moving the player, so this displacement is better now
- This is made through a custom general ability component which hasn't any ability it self, but it has event options for different ways to be used, like activate events on press down, press up or press hold with delay times for them
- So like the rest of abilities, the gravity control can be managed with just one key (which is the same to activate the rest of abilities, since only one of them is active at a time).
- This component allows to act as a bridge between any previous component and the ability system, to use any other feature as an ability
- Also, the gravity control was previously a little limited when using weapons or powers, but now, it can be used freely with them
- Added a new example to the ability system, this time, linking that ability (which is to push objects taking a position as center, like the player) with the action system, so the ability it self triggers a magic animation and with the delay option, the push objects function can

be activated, to check and push any object found in the radius. So this pretty much allows the option to configure spell casting and similar actions through the ability system
- Added cool down to each ability, so if the ability is used, a certain amount of time needs to pass before using it again
- Added time limit to each ability, so if the ability is active more than x time, this is deactivated
- Added the option to use a hot bar for the ability system (though currently there are only 4 slots, but any number can be configured) and I have used the arrow keys to use it as hot keys for it (but any keys can be configured for that). I have also added a time text for the time limit of the ability and a second image on each ability slot to show that cool down time until the ability is available again. Finally, the ability of each slot of the hot bar can be configured in game, just by opening the ability wheel menu, select an ability and press the button that you want to use for it
- Added another example of the action system combined with the ability system, to throw fire, including particles and damage area
- Added two more examples of abilities which use the action system to play animations, in this case, two more spell casts, one to electrocute with electricity (sending a remote event option to activate character's ragdolls) and other to place a fixed position heal area, in the form of rain, healing anything inside that area (in this case, the are is placed in the player's position but it is not moved from there, but the options allows to make that area to follow the player if you prefer or for other ability)
- Added last ability for the current update. It is used to "invoke" a meteor/missile from the air. It uses collision detection on the particle system to check when impacts on an object and activate the explosion
- Added a new action to throw objects, allowing to use a parable for it based on the player looking direction. It can be used for example to throw objects to call the attention of enemies in the level or even to damage them on impact (that can be disabled if you want to use it only for the first option)
- Added option to set the current ability of the player through the change player state system, used by trigger

GRAPPLING HOOK SYSTEM

- Added grappling hook system. Though in this case, it is more to be attracted to surfaces rather than swing around (that kind of movement will be added on the next update), taking the idea from doom eternal (the hook can be used of static surfaces or any rigidbodies, including enemies). It has multiple options for speed, camera and others and it is used through the ability system, though it could be used separately as well, like a feature always enabled (same for any other ability)

- Added option to rotate the player toward the target direction, use animations for the start and when a certain % of the total distance is reached, add down speed to simulate gravity. Also, fixed points for the hook to be attached, so if any of these points are on screen and with no obstacle between camera and the target, the system will take the closest point to the camera center. Also, the closest point has an icon over it, so it can be totally clear, similar to sekiro
- Configured an example of the current grappling hook system as attachment for weapons, in this case, for the double shotgun, in honor to doom eternal
- Added an option in the grappling hook system that allows to attract objects toward the player's position instead of pulling him towards them. Basically you just need to attach a component to the object to be attracted to identify it as a target to attract instead of pull. There are different options on that component, to trigger events and even to set a reduced speed value when is close enough to the player (similar to bulletstorm) giving time to the player to do something with it instead of pull that object to his face directly