movement

# robo escape /033

Walk LEFT/Right

# robo escape /033

Run LEFT/Right

# robo escape /033

Jump and Double

# robo escape /033

Wall Jump

# robo escape /033

DUCK

# robo escape /033

animation

# robo escape /033

FREEZE MOTION

# robo escape /033

camera

# robo escape /033

Health and damage system

# The Fire PRojectile mechanic robo escape /033

DEath zone

# robo escape /033

Staic Hazards

# robo escape /033

Dynamic hazards

# robo escape /033

NPC

# robo escape /033

attacking

# robo escape /033

Fire PRojectile

# The Fire PRojectile mechanic robo escape /033

Throw Gernade

# The Fire PRojectile mechanic robo escape /033

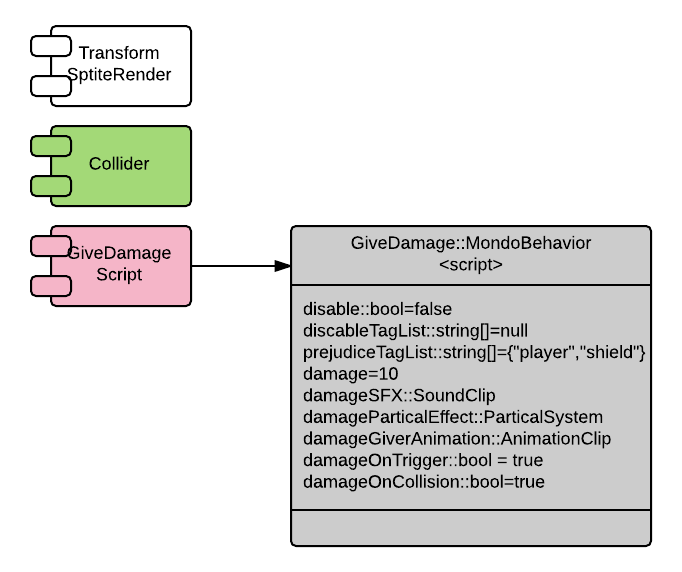
Mele

# The mele mechanic robo escape /033

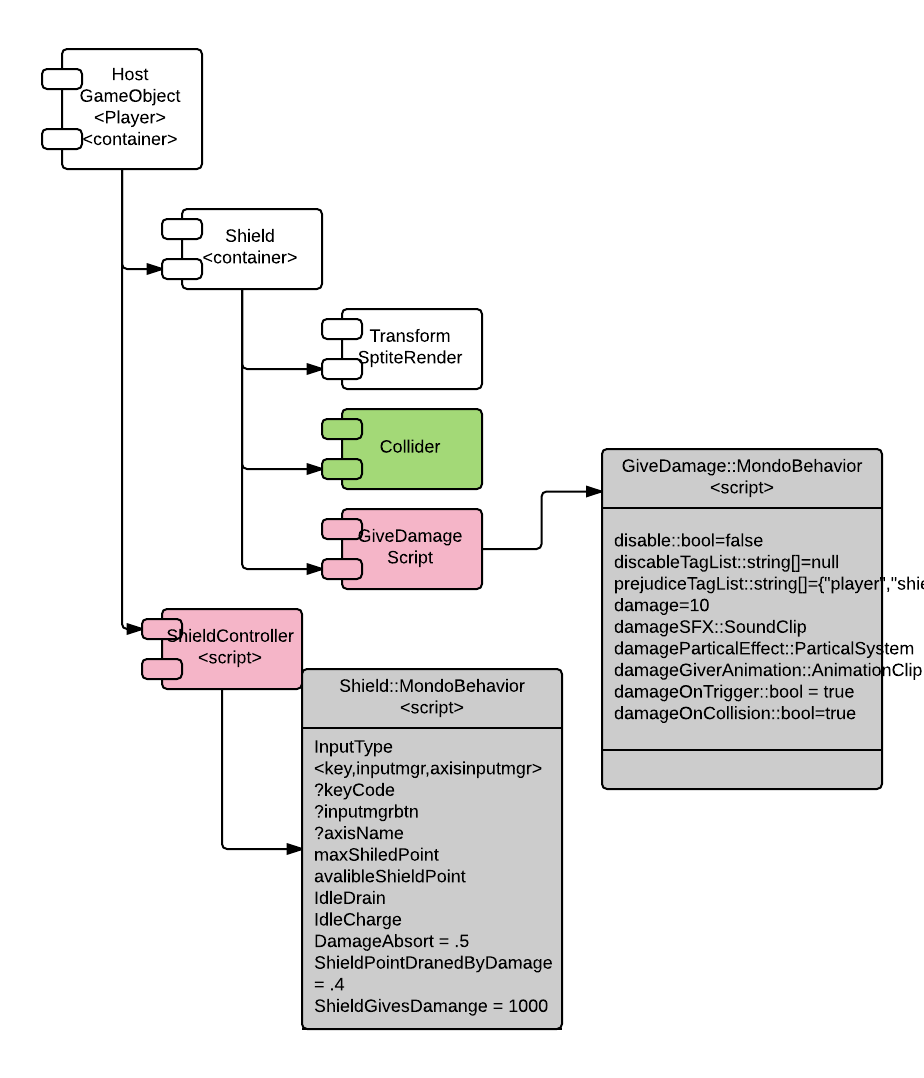
Shield

# The Shield mechanic robo escape /033

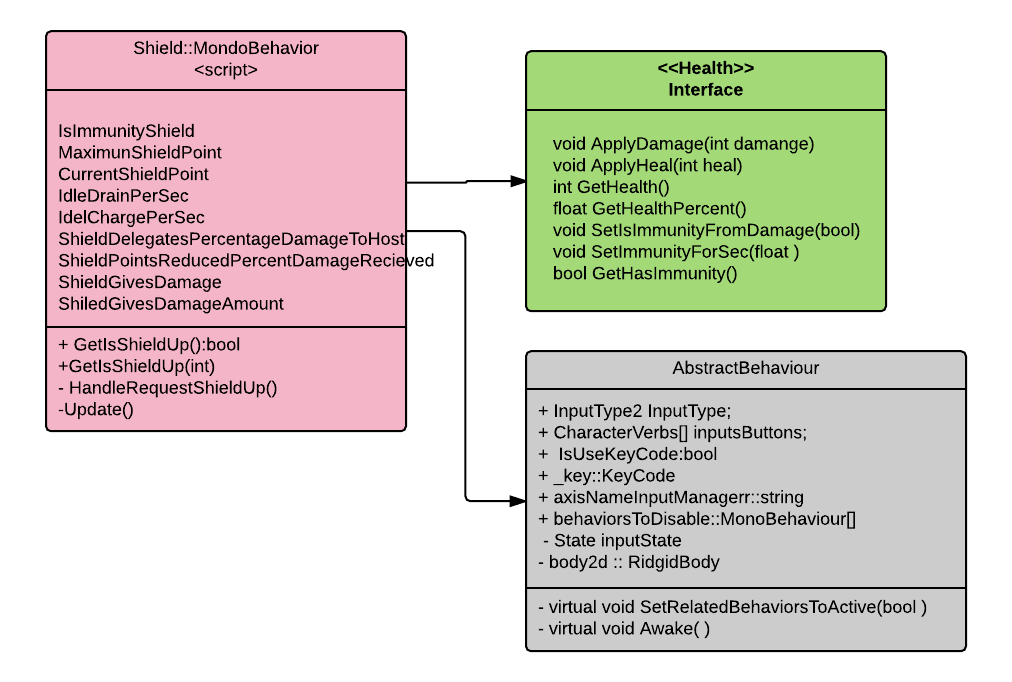
RoboEscape is a 2D platformer shooter. The shield in RoboEscape is facilitated by activating and deactivating a child game object on the hosting object. The child game object contained as a static collider, and a sprite render. Hence since the collider is not set to be a trigger object deflect with normal physics when applicable. The child object will be referred to as the shield. The shield game object host the GiveDamage script.



Since the shield is disabled by default we need the ShiledController to live on another object. The best alternative without re-engineering the simplicity of this system is the Hosting Game object, for example the player game object. I will remake that enabling and disabling the collider and sprite render is another alternative, that would have the benefit of being able to place the ShieldController on the shield game object; this have OO pros and cons.



The Game mechanic to support a versatile shield with upgrade and power up paths is difficult but sufficiently complex to warrant an explanation for clarity. The ShieldController has the following structure



The Shield Game Mechanic has the following requirement. The shields strength is represented by the CurrentShieldPoints. The act of having the shield active and Idle will drains the shield IdleDrainPerSec points. The shield will recharge when in active by IdleChargePerSec points. If the shield is marked as IsImmuniityShield the player will receive no pass-through damage from a damage deflection event. The shield will take a damage through the usual damage delegation system already in place since the shield conforms to the IHealth interface, the power of interface is demonstrative. The damage and heath subsystem will delegate a fractional damage to the host of the shield is the shield is not an Immunity Shield. The weight for the damage on the shield and the damage to the host are controller by the ShieldDelegatesPercentageDamageToHost and ShieldPointsReducedPercentDamageRecieved. Finally, the shield can be used as weapon if the shield has been designated as ShieldGivesDamage, and the damage yielded to an enemy is ShiledGivesDamageAmount. All the different option proved a very flexible shield behavior. The audio and visual effect are managed as usual on the DamageGiver and Health System.

Platform

# robo escape /033

MOving

# robo escape /033

oneway

# robo escape /033