

Advanced Phone and Smartwatch System

Smartphone System

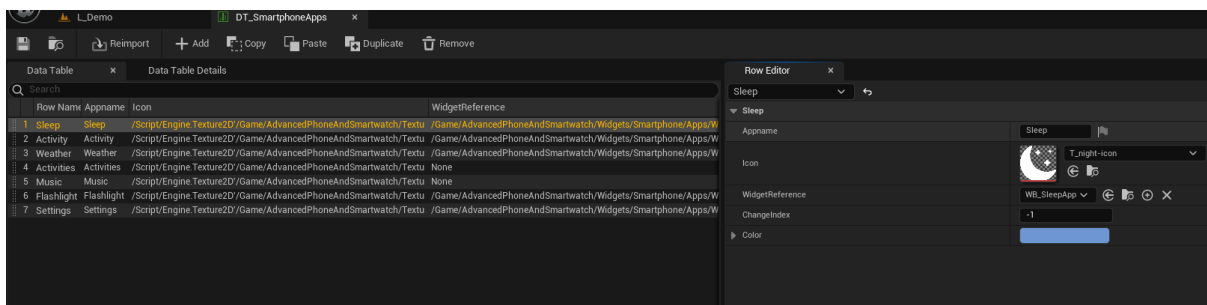
Integration

To integrate this project you need to use **BP_Pawn** as a main pawn in your Game Mode or copy all of the functions from BP_Pawn into your Character and redirect asset references (right mouse button on the content and FIX UP REDIRECTORS). To use time management you need to place BP_ExampleSky into your level. I didn't use any hardcasting, but as you can see in the reference viewer it's only selected in default game mode.

Adding and removing existing App

To add a new app – head to the DT_Apps and Add a new Row. You can set your Appname, icon texture, color of the background and optional change index if you want to use a widget switcher for it and/or widget reference that will be loaded. Remember that your widget should inherit from WB_BaseAppPayload to be visible in the DataTable.

To remove your application, simply press Remove in the DT_Apps and optionally remove the associated Widget that is in this row. In WB_BaseSmartwatch2DContainer you can optionally remove the associated index and create your own navigation with set ChangeIndex.



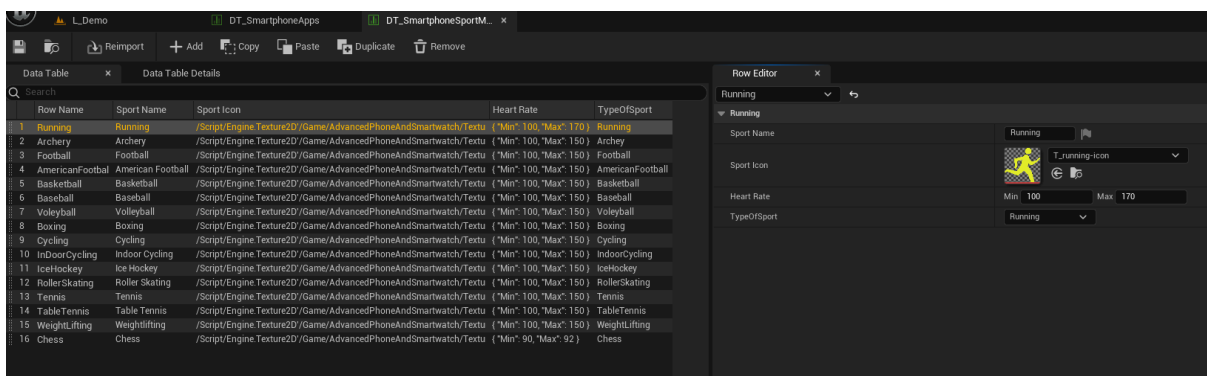
Structure

The main widget that we use is WB_BaseSmartWatch2DContainer. It contains the core that allows you dynamically assign any widget and functionality including ones with soft references that inherit from _WB_BaseAppPayload.

WB_Activity App that stores and calculates: Burnt calories that depends on sport type, our gender, weight, heart rate and age. We can adjust these statistics by changing the structure below:

- Some of the statistics are changed dynamically like burnt calories, steps, battery level and other dynamic statistics.
- Display statistics can be dynamically adjusted for desired format in smartwatch settings such as month-day format or using AM/PM.
- To add a new sport type and adjust heartbeat you can add your entry in **DT_SportActivities**. Activities included are:

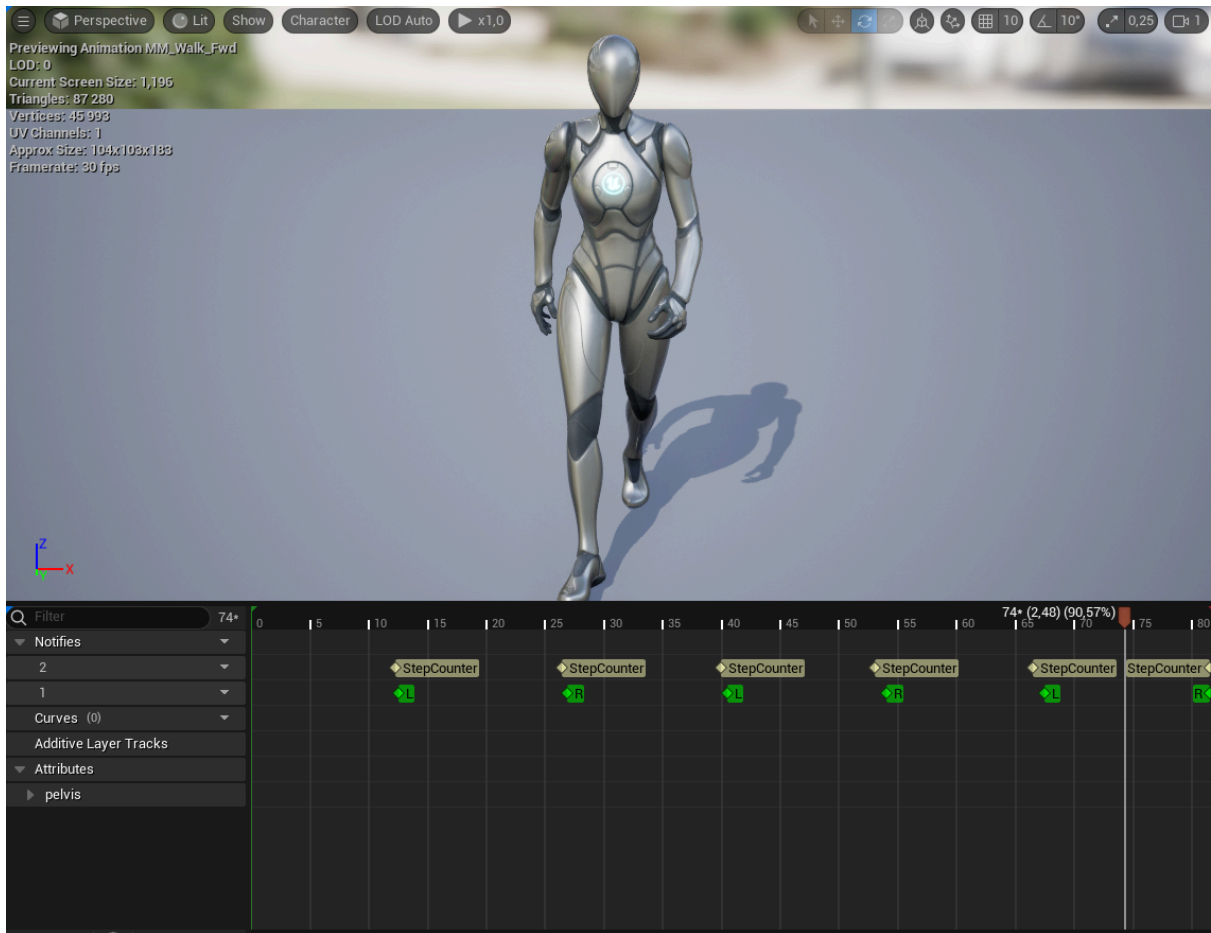
- Idle (default one)
- Running
- Football
- American Football
- Basketball
- Volleyball
- Boxing
- Cycling
- Indoor Cycling
- Ice Hockey
- Roller Skating
- Tennis
- Table Tennis
- Weightlifting
- Chess



Here you can add a new type of sport with its heart rate intervals.

Step Counter

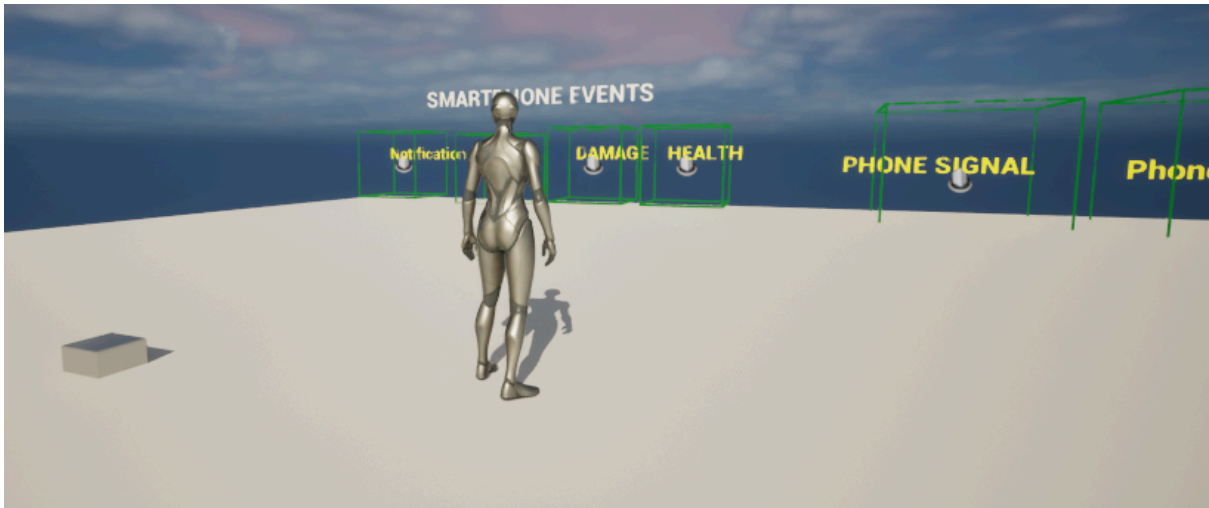
System includes a step counter that works on locomotion animations. All you need is to add this StepCounter notify to your custom animations



All we need is just a Step Counter notify in order to have our steps calculated in an app.

Other Notes

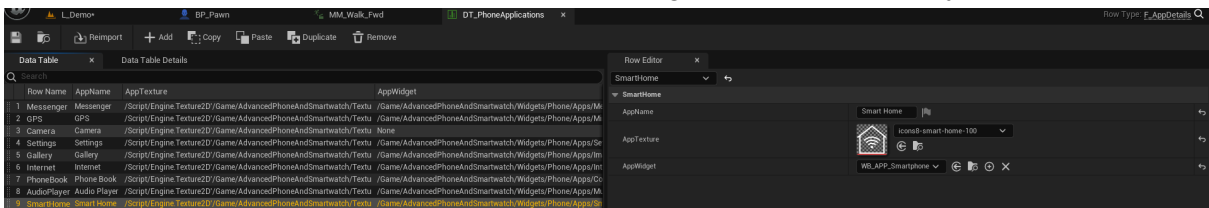
- There is a battery management that can lock your phone and example triggers that will add it (charge your device)
- There is an example health statistic that can send signals to your device without casting and update it on your smartphone
- Sprint system that calculates our heart rate with a stamina bar and regeneration system that will calculate burnt calories on it. This is just an example how you can use such systems in other sports that is based on dynamic variables
- There is a BP_ExampleSky that calculates time and for you and you can just change its format by simple setting in display
- There is versatile notification system and example trigger how it can be used gameplay wise
- Sample Death Screen widget to visualize gameplay incorporations
- Versatile base classes for each widget like Application Icons, Progress App Icons, Control icons, Misc icons for notifications



I've added a few example triggers to help visualize possibilities that are endless!

Phone System

Phone System Contains versatile applications that we can use for our games. All of these applications can be found in **DT_PhoneApplications** when we store: name of the app, icon of the app and soft reference for the application widget that can be displayed.



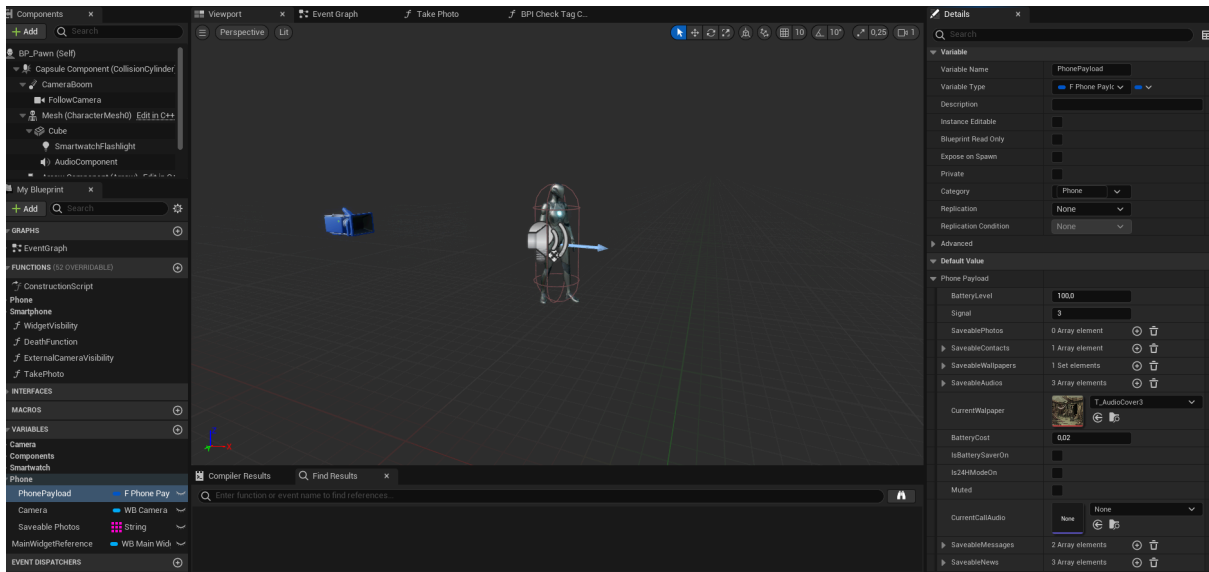
How to create a new App

We need to create a new widget that will inherit from **WB_BasePhoneApplication** and then add a row in the previous explain datatable where we set our icon, name and widget that we created.

Phone Payload

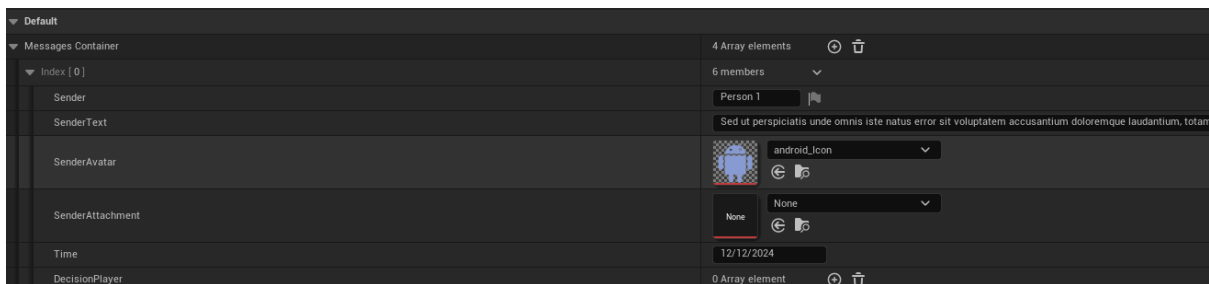
In BP_Pawn we can find a variable called **PhonePayload** where we can find variables that are dynamically changed

- Battery Level
- Signal (there is an example trigger how we can change signal based on our position in the world)
- Saveable Photos - we can make photos and then store them here and watch in the gallery)
- Saveable Wallpapers & Current Wallpaper - wallpaper management
- Battery Cost, BatterySaver, MutePhone, AM/PM - 24 Hrs Mode
- Saveable Messages - It's connected to a communicator
- Saveable News - It's connected to a web browser



Communication App

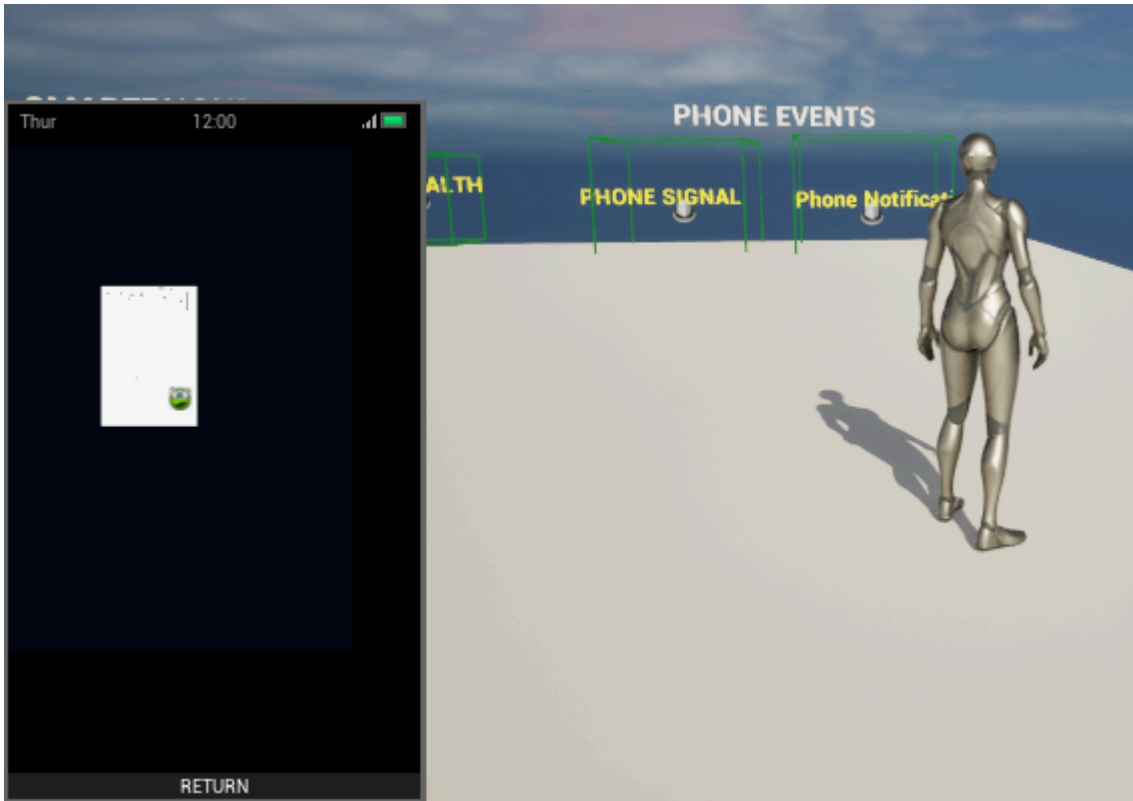
Allows us to create a group and dialogues that will be in this group. It's handled by a **BP_BaseDialogue**. I've provided two examples of dialogue communication that can be added to your PhonePayload in the main Pawn.



Here we can specify Sender name, Sender text, Sender Avatar, Sender Attachment (for now it's an image, I might add an recorded audio upon request), Time (can be dynamic with now function or set by us) and decision player where we can decide what to write in that group.

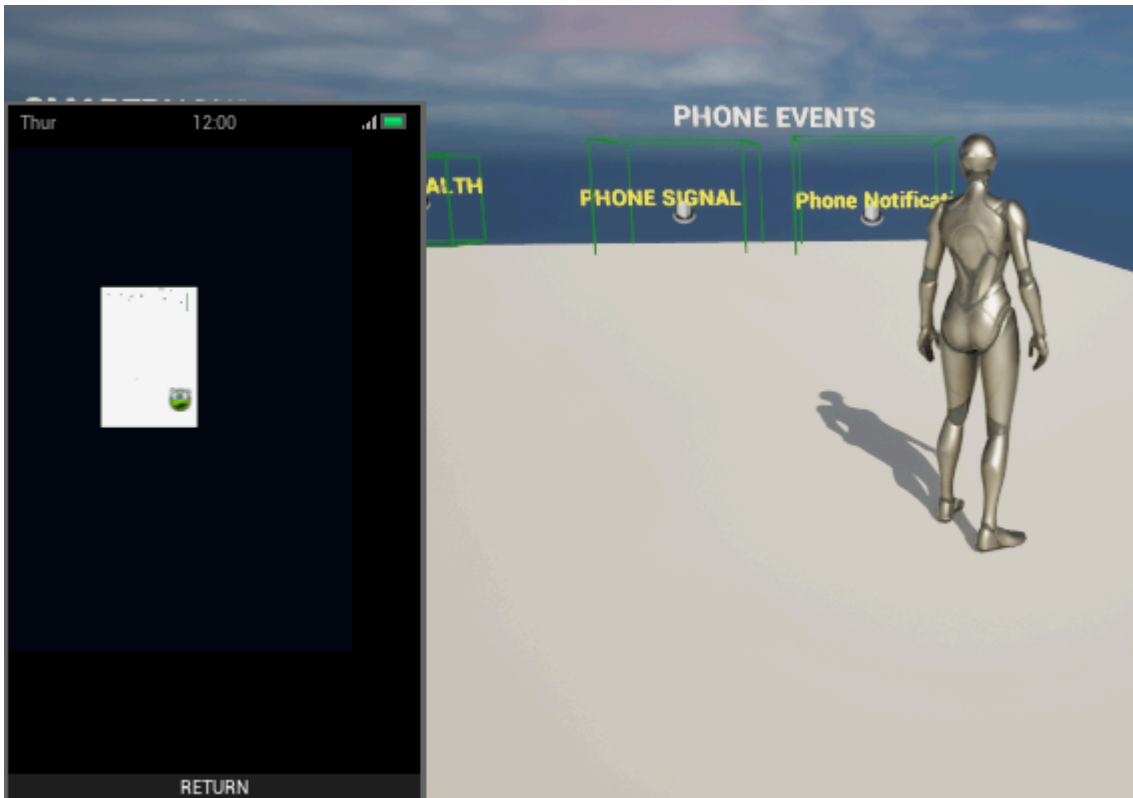
Minimap App

It displays our location and rotation in the real world. We can use it for a fancy quest system.



Camera App

Allows us to camera and take photos with LMB or quit camera mode with MMB. All of the pictures that we take will be visible in the Gallery App.



Gallery App

Here we can find our photo that we had taken a second ago. We can enlarge this photo by clicking on it.

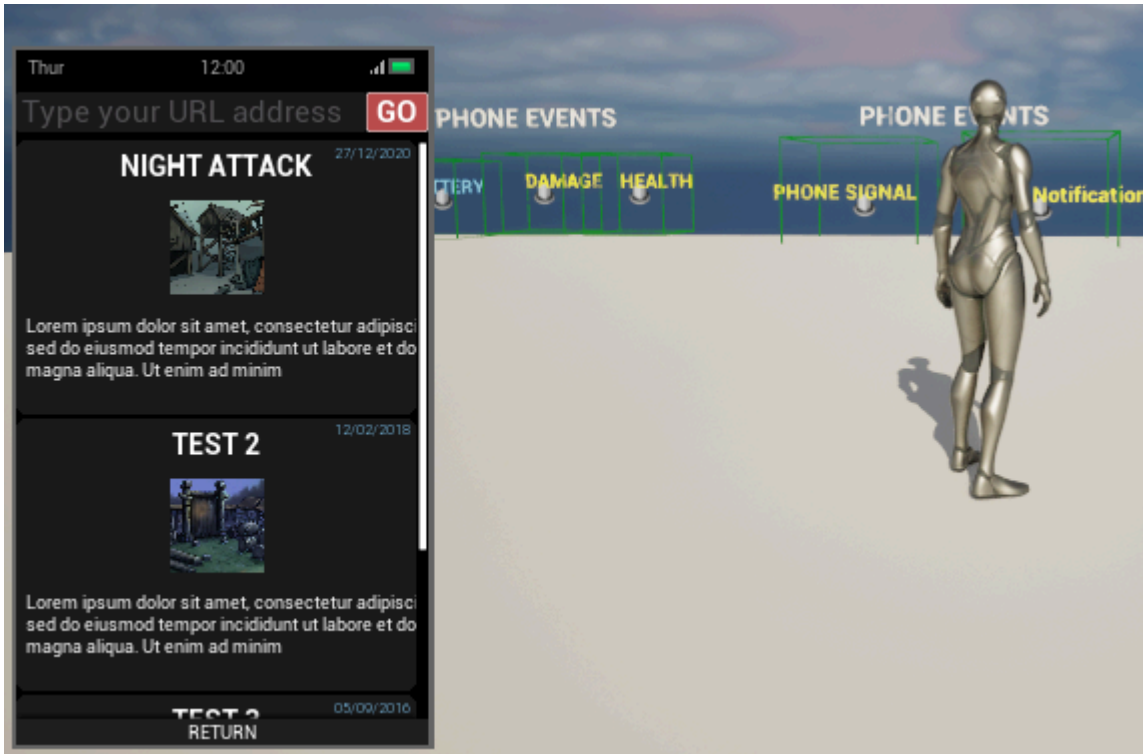


Settings App

Allows us to switch between 12/24 Hours modes, turn on Battery Saver (it will affect battery cost for the payload), and mute phone - we can use it for example a stealth game where incoming calls can attract enemies.

News App

Versatile app that allows users to dynamically add news on trigger and browse the internet. We can create a quest that will affect the world and dynamic news will be displayed conditionally.

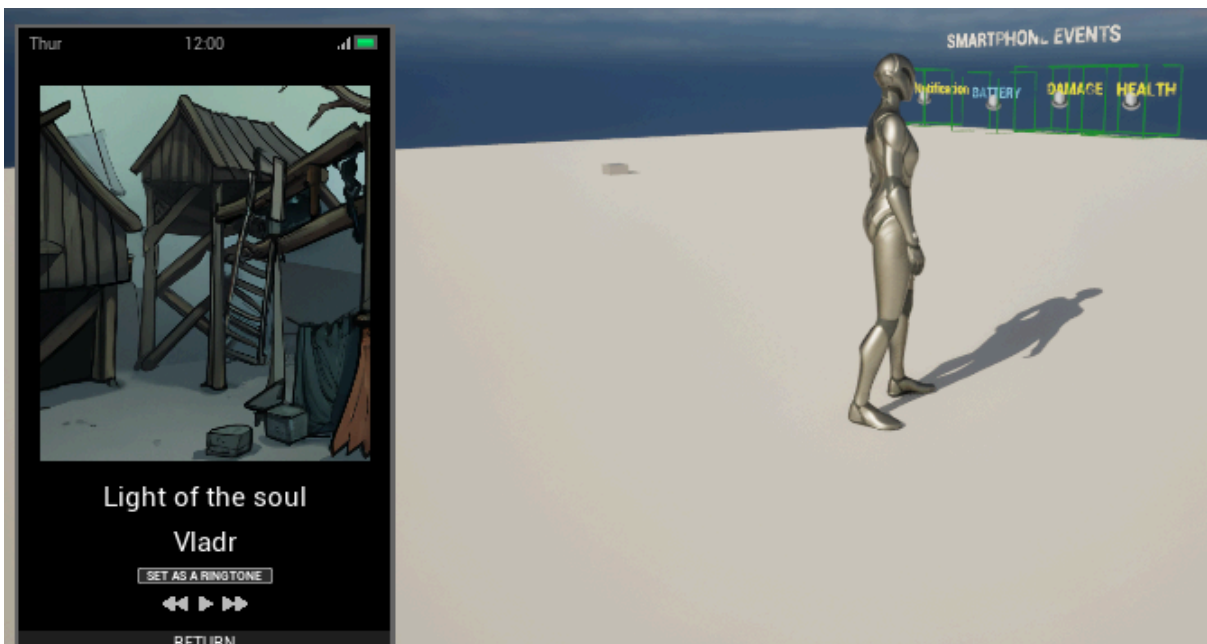


Contact App

It will allow us to add contacts dynamically and use functionality called update information about contact so we can dynamically change it there and use it as a glossary. You can use an event called **BPI_UpdateInfoAboutContact**

Music Player App

Versatile music player that allows us to play any type of music and set it as a ringtone.



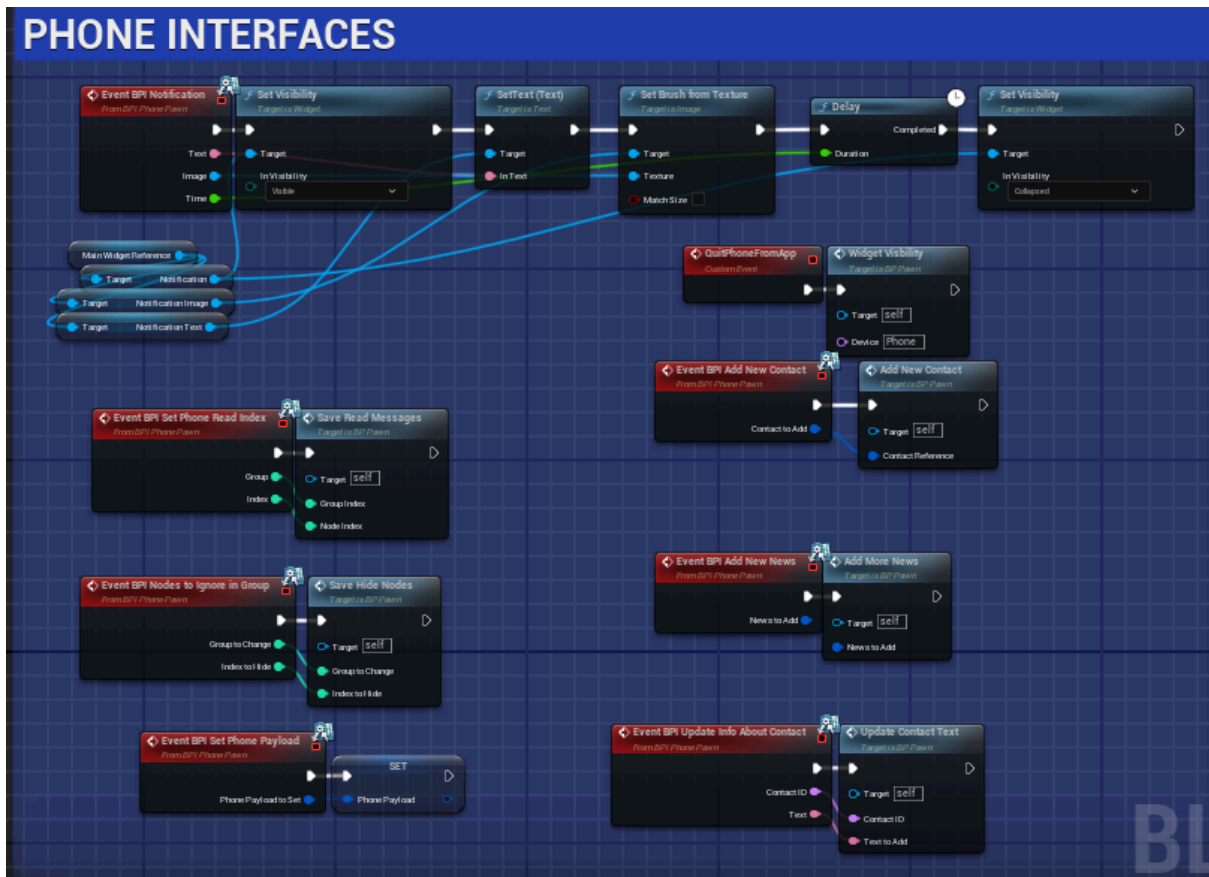
Smart Home App

Application that allows us to steer and manage functionalities outside of our phone. I've provided an example where we can manage BP_VacuumCleaner

Other News

You can find all of the related triggers in the Triggers folder and L_Demo map where I provided all of the examples that you can recreate yourself.

All of the events can be found under the Phone category tab in the **BP_Pawn** and Phone Interfaces in the event graph.



Feel free to ask me any questions!