BGS Technical Test

Controls:

W/LeftStick Forward - Push Up

A+D/Left Stick X Axis - Turn Left or Right

Space/ X Button (Dualshock) or A Button (Xbox) - Jump

Mouse/ Right Stick - Rotate Camera

Alt + F4 - Exit

Technical guide

The main actor in the project is **SKCharacter**. It is the character used by the player. This class is in charge of posses player visual representation in the game, contain our available **gameplay actions** and trigger it through user input. This gameplay actions are coordinated in conjuntion with skeletal mesh's anim blueprint (**AnimBP**) to sincronize it with our animations though **anim notifies**. This notifies are set in our animation and tell SKCharacter when it should jump or accelerate. For that reason we have two custom anim notifies: **SKJumpAnimNotify** and **SKSpeedUpAnimNotify**. Those animation are managed by an **State Machine** (Locomotion) with data extracted from SKCharacter through AnimBP's event graph.

One gameplay action that are not managed by SKCharacter nor AnimBP is the detection of obstacles. The SKCharacter contains a Obstacle Detection Component (**SKObstacleDetectionComponent**) which notifies the gamemode when we has jumped over an obstacle. *An obstacle was jumped when we are flying over an actor from the beginning to the end of its size dimentions*. We determine this by checking if an actor enters and exits from underneath the skateboard with a box collision. When an actor success with this condition, the player gains points.

The **SKGameMode** receive this event and grant score to user's player state (**SKPlayerState**), which notifies of this to the UI (a **SKScoreWidget** spawned by **SKHUD**).

Final notes

My first step to face this test was to investigate about the Tony Hawk Pro Skater saga by mostly examinating gameplay of THPS 1+2, paying attention to the requested mechanics. I noticed that the pushing functionality was different from what I expect and I would have preferred to implement that version but the lack of detail about this mechanic in the Doc makes me doubt, so I decided to do a more intuitive approach (yet I understand the design reason behind the push up of THPS).

I also decided to invest a lot of time in polishing since I understood that it is an important aspect of this test and more importantly, for the studio.

To conclude, I occupied a total of ~18 hours of full work time from initial gender research to the packaging of a build.

- 2 hours of game genre and technical investigation.
- 1 hour invested on setup the project (create default UE project, create git repo and upload it to GitHub, cleanup default template stuff, tweak project settings, etc.).

- 3 hours to implement base movement with default assets as placeholders.
- 2 hours to get all the assets.
- A total of 6 hours to import and polish all the game assets (most of the time spent of fixing animations and modifying the final map).
- 2 hour to implement obstacle detection and to polishing it.
- 2 hour of general final polish (project assets organization, code cleanup, gameplay tweaks).