David Vader
davevad
EECS494

Project 2 Research & Analysis

Temple Run

The first game I will be totally-not-ripping-my-game-off-of is Temple Run, everyone's favorite mobile addiction (for that one year). Temple Run is simple to describe, you control a Lara Croft, Indiana Jones-type explorer, and are tasked *immediately* with escaping a horde of what look like shadow monkeys. However, in this temple exploring game, you permanently run in a forward direction, and swiping left or right on the screen with turn / slide / jump your character that direction, and sometimes you can jump over obstacles and pickup coins and power-ups. So all-in-all it is the usual recipe, however, TR hit many things right on the money, but most importantly: JUICE. This aspect is what I want to emphasize in my implementation of an "infinite-runner".

If you have played Temple Run you have seen and felt exactly what I'm talking about. The menus in Temple Run are simple, just like any mobile game. Examining closer, what they do is very clever. They do not use a separate screen as the start menu, and load in the game when you touch START. Instead they put you right into the game, pause before they introduce the hero, and load the start menus in, overlapping the environment of the game-start-intro. That is genius! Every other game has a loading screen, then a start menu, then a loading screen, then level 1. Putting the player into the game, and having this seamless transition from menu to play is the juice that can put a game on the leaderboards. Even when you lose the game, the game over buttons and graphics fall into place, with a sense of gravity, and bounce when they arrive. Its subtle craftsmanship like this that makes your eyes just… happy!

In my own implementation I want to emphasize this type of juice, where things just feel, and look…. right! I want to create a flow of start-to-finish such as Temple Run, where loading into the game already feels like you're playing, and you can enter a "run" of the game without feeling a cut, or a halt, like I am waiting for something to load. This ease from menu to gameplay, and then game over menu back to gameplay will be a large focus for my game, because a fluid, soft on the eyes, and aesthetically pleasing transition from failure to gameplay is exactly what will make a player hit retry twice, and hopefully more.

Bit Trip Runner

The next game I'll be examining is Bit Trip Runner, another infinite-runner, but this one slightly different, and slightly less infinite.

BTR is a platformer that forces the runner, you, to always run to the right, and gives you control over jumping and sliding. But with simplicity, comes great power! You jump over pits and enemies, and slide through destructibles, you collect coins and power-ups, and try to do the best before you reach the end of the level. So again, the same recipe. However BTR has you strangely addicted in a matter of seconds, which I believe is greatly due to the simplicity, and the accessibility.

By accessibility, I mean the game is easy to pick up, easy to learn, but difficult, and extremely rewarding to master. The game introduces you to mechanics rather well, creating a good sense of difficulty scaling, like any good game. However, BTR forges their gameplay around keeping the controls somewhat limited, by never introducing more mechanics than new things to jump on, or cooler objects to slide through / under. The music seems to wrap around how the player is moving and jumping, and all of the environment seems to react with juice as you run by. Games that are too overbearing with convoluted mechanics, and what I bet is very awesome C# code, can leave the player feeling too overwhelmed. Where as a game that is meant to be simple, like an infinite runner, should keep their controls light, and try to make the few options that they actually give the player feel juicy and addicting to try and master.

Simplicity is a complex quality in itself, because the less you overbear your game with, the harder a focus the player will have on your few mechanics. A game with very few mechanics, addicting controls, and a TON of juice, will have players glued to the screen, retrying level 3 87 times until they can get all 50 stars in one run. In my implementation I want to focus on keeping controls lightweight, and to focus on the simplicity of an idea rather than the complexity. My goal is to center an addicting atmosphere based on conveyor belt reaction gameplay, much like BTR, but with a competitive twist. A second player would control the environment, so the enemies and obstacles. Player 2 would also with a limited control palette, to keep strategy simple and accessible, but with the ability to master and outsmart Player 1, the infinite running hero. In this way, I could create the addicting, keep-hitting-retry, gameplay that has gamers hooked on games like Temple Run and Bit Trip Runner, but with a new twist!