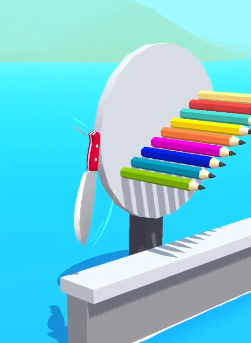
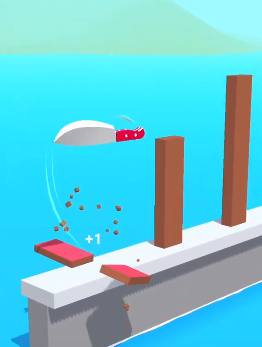
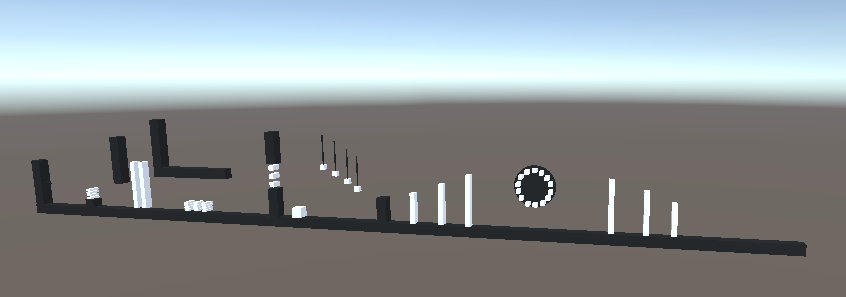
Relatório de desenvolvimento Clone Slice It All

Felipe Vergara

Dia 1  
 Em um primeiro momento, antes de iniciar a produção de um protótipo Clone de Slice It All, baixei o jogo e joguei ele por algum tempo, analisando as suas mecânicas e anotando suas singularidades para então ver qual grau de semelhança as mecânicas que seriam desenvolvidas teriam com as já existentes e ter ideias de como desenvolvê-las.

* **Movimentação**  
   No jogo original a movimentação da faca ao clicar é na diagonal, jogando ela pra cima enquanto ela é rotacionada em alta velocidade até o ponto que ela fica na horizontal. No clone, para um desenvolvimento mais rápido foi decidido que, ao clicar, a faca apenas iria ir para cima enquanto rotacionava, a movimentação na horizontal viria da fase, que iria para a esquerda, e só pararia de se movimentar caso o jogador colidisse com o cenário.  
   Para desenvolver a movimentação do pulo, inicialmente seria projetado usando o Transform.Translate para testes, e depois seria trocado para o Rigidbody.MovePosition. Para a movimentação da fase seria usado o Tranform.Translate
* **Rotação** No jogo original, independente do momento que você clicar, a faca rotaciona até o ponto que ela fique na horizontal. E caso você não clique, ela cairá lentamente enquanto rotaciona devagar, até o ponto que ela fica de cabeça para baixo então a rotação é acelerada automaticamente. No clone essas características foram mantidas.  
   Para desenvolver a rotação, inicialmente seria projetado usando o Transform.RotateAround, para testes, e depois seria trocado para o Rigidbody.MoveRotation.
* **Destruíveis** No jogo original, quando a faca encosta nos objetos eles são cortados, e para recriar isso foi pensado em duas hipóteses, a primeira é que os Mesh’s dos objetos seriam cortados, porém como os objetos sempre eram cortados da mesma forma, velocidade e posição quando atingidos, essa ideia foi descartada, ficando com a hipótese de que seriam dois GameObjects colados e travados, e quando a faca encostasse neles, eles apenas seriam “soltos”.  
   Para desenvolver o plano era pegar um cubo da Unity e reduzir o tamanho dele até que parecesse que fosse metade de um cubo, então pegaria outro cubo idêntico ao primeiro e colocado ao lado dele, criando a ilusão de que é apenas um objeto. Para ativar o corte, seria adicionado colliders(triggers), rigidbodys e tags nos objetos para simular o efeito de corte e de queda.  
   O posicionamento de alguns objetos foram baseados nos posicionamentos dos objetos do jogo principal, como a pilha de objetos, a roleta de objetos e os objetos com tamanhos gradativos.
* **Paradas**  
   No jogo original, quando o jogador atinge o chão, paredes ou objetos com a parte do cabo da faca, ele é penalizado sendo jogado para trás e sofrendo um pequeno stun, sendo impedido de rotacionar a faca por algum tempo. Caso ele atinga o chão com a lâmina da faca, ela fica infincada, parando a faca, que pode voltar a rotacionar caso o jogador clique novamente na tela. No clone essas características foram mantidas  
   Para desenvolver as mecânicas seria adicionada colliders, tags e modificar variáveis para controlar os efeitos, também seria adicionado uma corrotina para controlar o stun.
* **LevelDesign** No jogo original existem diversos tipos de níveis que pode ser jogado, como Campo, Cidade, Floresta e Oceano, porém como estava sendo desenvolvido um protótipo simples o Level seria apenas monocromático simples, com cores que combinassem com a faca. A montagem do Level foi desenvolvida baseada no jogo Flappy Bird e no jogo original.

****

**Dia 2**

Com as ideias principais já idealizadas, foi iniciado o desenvolvimento inicial para ver se as ideias pensadas no dia anterior funcionariam e trariam a lembrança do jogo original.

* **Movimentação e Rotação**  
   A implementação da movimentação e da rotação deram certo, porém foi preciso algum tempo até conseguir encontrar valores de rotação e pulo que coincidiam para que, a cada clique, desse uma volta inteira e voltasse ao mesmo ponto inicial quando caísse. Para o controle das etapas de rotação foi usado o transform.localEulerAngles.x para verificar se estava na posição correta para a velocidade de rotação.  
   A movimentação da fase também foi implementada usando um Transform.Translate, e não foi necessário trocar para um Rigidbody.MovePosition.
* **Destruíveis** Para testar se os objetos estavam sendo destruídos da maneira correta, foi necessário adicionar os Rigidbodys e então configurar para que, se encostasse na lâmina, o objeto perderia sem parentesco, ativaria sua gravidade e seria suavemente jogado para a lateral, caindo, poucos segundos depois, quando já estivesse fora da visão do Jogador, seria destruído.   
   Para conseguir os efeitos e não atrapalhar os cálculos da movimentação, foi necessário mudar diversos números da física da Unity, tendo que criar cálculos de gravidade própria para a faca, modificar o Contact Pairs Mode da Unity para conseguir usar o modo Is Kinematic dos Rigidbody. Foi um pouco problemático para encontrar números que combinassem, porém nada muito trabalhoso.

**Dia 3** Com o projeto praticamente funcionando, foi necessário apenas trocar o método de movimentação e rotação da faca, arrumar bugs simples de controle da rotação automática, criar um Canvas com pontuação e uma forma de encerar a fase.

* **Movimentação e Rotação** A troca de tipo de movimentação e rotação foi simples e resolveu a maioria dos bugs que o protótipo tinha, deixando a movimentação mais fluida e estável, os números da nova movimentação e rotação foram os mesmos da antiga, economizando tempo.
* **Bugs**  
   A maioria dos bugs foi arrumada ao trocar o tipo de movimentação, e os que se mantiveram foi necessário apenas adicionar alguns controles das variáveis de controle em algumas linhas que tinham sido esquecidas e não estavam permitindo um loop de Gameplay correto.
* **Canvas** Um canvas simples foi criado para mostra uma pontuação e uma tela de vitória.   
   A pontuação é simples e simplesmente é adicionado 1 a pontuação quando um objeto é cortado.  
   A tela de vitória é ativada quando o jogador termina o percurso do protótipo, e nela tem um texto parabenizando o jogador e um botão para reiniciar a fase.

Com o protótipo aparentemente totalmente funcional, foi feito o APK para testar no celular sem a necessidade do UnityRemote, confirmado o funcionamento do projeto o projeto foi finalizado.