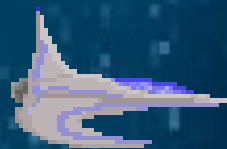


# Game Design Document

## Apollo Runners



# Sumário

Game Design Document.....	4
2. Visão Geral.....	5
2.1. Conceito do Jogo .....	5
2.2. Gênero .....	5
2.3. Público Alvo.....	5
2.4. Resumo do Fluxo de Jogo .....	5
2.4.1. Como o jogador se movimenta pela interface .....	5
2.4.2. Como o jogador se movimenta pelo jogo .....	5
2.4.3. Persona dos jogadores .....	5
3. Jogabilidade e Mecânicas.....	6
3.1. Jogabilidade .....	6
3.1.1. Progressão do Jogo .....	6
3.1.2. Estrutura de Missões / Desafios.....	7
3.1.3. Estrutura de Enigmas .....	7
3.1.4. Objetivos .....	7
3.1.5. Fluxo do Jogo.....	7
3.2. Mecânicas .....	8
3.2.1. Física .....	8
3.2.2. Movimento do Jogo .....	8
3.2.3. Objetos .....	8
3.2.4. Ações .....	8
3.2.5. Combate .....	8
3.2.6. Economia.....	9
3.2.7. Fluxo de Telas .....	9
3.3. Opções do Jogo .....	11
3.4. Rejogo e Armazenamento .....	11
3.5. Trapaças e Descobertas.....	11
4. História, Configuração e Personagens .....	11
4.1. História e Narrativa .....	11
4.2. Mundo do Jogo .....	11

4.2.1. Visão Geral do Mundo.....	12
4.2.2. Áreas.....	12
4.3. Personagens.....	12
5. Fases / Níveis.....	12
5.1. Fase Base .....	12
5.2. Fase de Treinamento .....	12
6. Interface (imagens).....	12
6.1. Sistema Visual.....	12
6.2. Sistema de Controle.....	13
6.3. Áudios, Músicas e Efeitos Sonoros.....	13
6.4. Sistema de Ajuda.....	13
7. Oponentes .....	14
7.1. Inteligência de Oponente / Inimigo .....	14
7.2. Personagens Amigáveis e de Não-Combate.....	14
7.3. Inteligência de Suporte .....	14
8. Requisitos Técnicos .....	14
8.1. Hardware Alvo.....	14
8.2. Hardware e Software de Desenvolvimento.....	15
8.3. Requisitos de Rede .....	15
9. Arte do Jogo.....	15
9.1. Design Inicial.....	15
9.2. Sprites .....	17
9.3. Partículas.....	18
9.3.1 Background.....	18
9.3.2. Habilidades.....	18
9.3.3. Propulsores.....	19
9.4. Skybox .....	20
9.5. Programação.....	21
9.6. Recursos Adicionais .....	21
10. Game Design Canvas .....	23

# Game Design Document

## Apollo Runners

Um jogo divertido de corrida espacial



# GitHub

Integrantes:

Beatriz Urgal  
Nícolás Lúcio  
Nicolau Isaac

## **2. Visão Geral**

Apollo Runners é um jogo de corrida entre naves espaciais, tem o tema focado no espaço sideral e é todo desenvolvido em pixel art.

### **2.1. Conceito do Jogo**

O jogo se baseia nos simples jogos de corrida na época auge do Arcade, tendo mecânicas simples e igualitárias de movimentação e utilização de habilidades adicionais. Visando apenas uma diversão rápida e simples para os saudosistas do gênero.

### **2.2. Gênero**

Corrida espacial e casual.

### **2.3. Público Alvo**

Jovens e adultos que procuram uma distração rápida com um desafio simples.

### **2.4. Resumo do Fluxo de Jogo**

O jogo se resume em três telas de simples acessos, sendo as seguintes. Menu principal, onde o jogador pode iniciar o jogo e/ou ver uma básica tela de tutorial com os comandos básico do jogo. A tela do jogo, onde toda o desafio acontece, e uma simples tela de vitória mostrando qual player saiu vitorioso.

#### **2.4.1. Como o jogador se movimenta pela interface**

Ele se movimenta pelos direcionais básicos do teclado do computador, sendo ou as teclas de “WASD”, ou as setas direcionais localizado ao lado direito do teclado.

Ao ser jogado pelo Arcade, que é a plataforma alvo para o jogo, a interface pode ser navegada por ambos os eixos direcionais localizados na estrutura do Arcade.

#### **2.4.2. Como o jogador se movimenta pelo jogo**

O jogo se baseia em um conceito básico de movimentação, que consiste nos seguintes:

Para acelerar, o jogador deve mover seu eixo de movimentação para a direita, caso ele esteja no Arcade, é só mover o Joystick de eixo para a direita, caso seja no computador, ele utiliza do botão “D”.

Para se movimentar para cima e para baixo com a nave, é obrigatório que o player já esteja acelerando, sendo assim, as condições para isso requerem que no Arcade, o eixo seja apontando para a direcional superior caso o player queira subir, ou para a direcional inferior caso o player queira descer.

Caso ele esteja jogando no computador, se resume em o player apertar o “W” para subir, e o “S” para descer, mas lembrando que é obrigatório que o “D” esteja pressionado junto para ambas as situações.

#### **2.4.3. Persona dos jogadores**

Na etapa de desenvolvimento, foram designadas três personas básicas, que são:

#### Persona Primária 1:

- Nome: Rick Arc do Milos.
- Idade: 16 anos.
- Gênero: Masculino.
- Escolaridade: Ensino Médio (cursando).
- Plataformas Preferidas: Console e computador.
- Objetivos Preferidos: Correr, capturar e perseguir.
- Tipo de Jogador: Conquistador.
- Principais Atrativos: Competição entre ele e seus amigos do colégio, e com o tema do jogo, já que ele quer ser um astronauta quando crescer, e é apaixonado pela NASA.

#### Personagem Primária 2:

- Nome: Evelyn Melo.
- Idade: 24 anos.
- Gênero: Feminino.
- Escolaridade: Ensino Superior (concluído).
- Plataformas Preferidas: Computador e mobile.
- Objetivos Preferidos: Capturar, ação proibida e perseguir.
- Tipo de Jogador: Assassino.
- Principais Atrativos: Usufruir das mecânicas do jogo para atrapalhar o progresso do adversário, e também a simplicidade das mecânicas e objetivos do jogo para que ela jogue até quando estiver incapacitada de seus reflexos e sentidos.

#### Persona Secundária:

- Nome: Cláudio Isbrand.
- Idade: 60 anos.
- Gênero: Masculino.
- Escolaridade: Ensino Médio (concluído).
- Plataforma Preferidas: Jogos de tabuleiro e cartas.
- Objetivos Preferidos: Conhecimento e solução.
- Tipo de Jogador: Conquistador e socializador.
- Principais Atrativos: O uso da estratégia para chegar no seu objetivo antes do oponente com o uso de Power Up, e a interação entre ele e o seu oponente, tanto dentro do jogo como do lado da mesa do Arcade.

### 3. Jogabilidade e Mecânicas

Este tópico explicará mais detalhadamente sobre os quesitos de jogabilidade e mecânicas encontradas durante o jogo.

#### 3.1. Jogabilidade

##### 3.1.1. Progressão do Jogo

O jogo consiste em uma corrida, que se define em algo chamado “melhor de três”, que se resume em uma partida de três partes, cada parte se finalizará com a vitória de algum dos dois players. Ao ganhar, o player ganha um ponto no sistema de corrida, ao conseguir dois pontos, o player possuidor deles ganha a partida. Logo, caso haja empate de um ponto para cada, ganha aquele que concluir a próxima partida, se finalizando assim o sistema de “três partes”.

Assim que concluído a parte acima, os jogadores são levados para a tela de vitória.

### **3.1.2. Estrutura de Missões / Desafios**

O jogo não tem missões e/ou desafios adicionais, apenas os objetivos básicos para vitória da corrida, que serão explicados futuramente neste tópico.

### **3.1.3. Estrutura de Enigmas**

O jogo não possui de mecânicas e/ou jogabilidade de enigmas.

### **3.1.4. Objetivos**

O único objetivo do jogo, é ganhar a corrida, para isso se segue os seguintes quesitos:

A jogabilidade se resume na distância que um player está do outro, essa distância será a partir deste ponto chamada de “Unidades”. Quando um player atinge uma certa Unidade de distância do outro player, ele ganha um ponto para o cálculo de vitória da partida.

Após conseguir dois pontos, assim como citado na parte “Progressão do Jogo” deste documento, o player que atingiu tal objetivo, ganha o jogo.

### **3.1.5. Fluxo do Jogo**

Na visão dos jogadores, ele terá acesso à seguintes informações:

A velocidade da sua nave, ela será baseada apenas no cálculo em quilômetros por hora (Km/h) da nave no momento atual da gameplay.

O tempo para a partida, contada de forma decrescente, que serve para uma mecânica que será explicada posteriormente.

A distância em que o player se encontra do outro player, para poder atingir o objetivo citado no tópico “Objetivos” acima.

E qual habilidade ele pode utilizar ou quanto tempo precisa esperar para conseguir a sua próxima habilidade, que serão devidamente explicadas no tópico “Combate”.

O fluxo do jogo se resume inicialmente na conclusão do objetivo, mas ele pode ser levemente alterado caso esse objetivo não seja concluído rapidamente, assim, se aplica o conceito de tempo de partida.

Cada começo de partida, o relógio se reseta para a marca de dois minutos, e a partir do seu início, ele marca o tempo da partida de forma decrescente, até a marca zero do timer.

Quando o relógio atinge metade do seu tempo, com a marca de um minuto, se inicia o sistema de Morte Súbita do jogo. Esse sistema se consiste em, a cada segundo passado abaixo de um minuto, a Unidade necessária para vitória da partida se reduz em 10%.

Por exemplo, caso no segundo anterior, era necessário 1000 (mil) Unidades de distância para a vitória do jogador, no próximo segundo será necessário apenas 900 (novecentas) Unidades de distância para vencer.

Ao atingir a marca de zero segundos, quem estiver na frente, independente da distância, ganha a partida.

## **3.2. Mecânicas**

### **3.2.1. Física**

Todas as físicas desse jogo foram calculadas manualmente para que seja possível simular um ambiente sem gravidade e espacial para as naves dos jogadores. Sendo assim, apenas a força inicial se é calculada para um impulso primário, qualquer ação posterior terá sua própria reação programada de acordo com o ambiente que o jogo se passa.

### **3.2.2. Movimento do Jogo**

Os movimentos básicos, em quesitos gameplay, já foram citados no tópico “Como o jogador se movimenta pelo jogo”, sendo assim, se segue as explicações adicionais sobre o mesmo.

Sempre quando o player acelera a sua nave, uma força será calculada para simular o seu impulso se baseando no seu coeficiente de aceleração.

A aceleração inicial, quando se mantém a nave acelerando, diminui progressivamente para poder assim calculando a velocidade atual da nave. Quando a nave é desacelerada, esse coeficiente volta a crescer até atingir o ponto original de partida da nave.

Esse sistema impede que uma nave que já está acelerando tenha muita vantagem, e para que caso alguma nave perca a sua aceleração ou seja obrigada a parar, consiga voltar à partida de forma justa.

Os outros elementos do jogo são iniciados com apenas físicas iniciais, seguindo uma força linear até atingir um objeto. A partir desse momento, a força resultante irá fazer com que um dos dois objetos seja prejudicado de sua trajetória base.

### **3.2.3. Objetos**

Todos os objetos adicionais, serão criados a partir da nave do player, sendo ele correlacionados ao sistema de habilidades do jogo.

Para adquirir ele, se deve basear no sistema de tempo para conseguir a próxima habilidade, e assim poder usá-la, como citado no tópico “Fluxo de Jogo”.

### **3.2.4. Ações**

As ações básicas de movimento já foram citadas acima, cabendo apenas às ações adicionais serem explicadas.

O player será capaz de resetar a rotação da sua nave, caso o mesmo perca o controle dela. Mas essa ação só pode ser feita caso o player pare de acelerar a sua nave.

Sempre quando uma habilidade estiver disponível, o player pode optar por utilizá-la, lembrando que ele só conseguirá uma nova caso use a anterior.

Todos os detalhes sobre a utilização de essas ações em nível de jogador, serão devidamente explicadas no tópico “Interface”.

### **3.2.5. Combate**

O conflito do jogo se consiste em atrapalhar o outro jogador, para isso, existe duas habilidades designadas a este propósito.

A primeira, chamada de “Beacon” no jogo, se baseia em um tiro que sai da nave que o optou por utilizar, esse tiro sairá em alta velocidade, em linha reta, para frente da nave.



Caso esse tiro atinja o outro player, o mesmo o fará que perca o controle, seja empurrado, e perca a velocidade da nave.

Logo, devido a todos esses quesitos, existe a dificuldade de se atingir o tiro, o que torna necessário relatar os seguintes balanceamento do mesmo.

Ao início do tiro, ele terá uma velocidade base inferior comparado ao resto da sua trajetória, e também um tamanho de acerto menor. Isso tornará mais difícil ser acertado em curta distância, em contramedida, ele fica cada vez mais rápido e maior de acordo com a sua trajetória, para contestar o fato de ser mais fácil para o outro player desviar para cima e para baixo dependendo da sua distância.

A segunda habilidade, chamada de “Barrier” no jogo, se baseia em barreira que sai da nave que a optou por utilizar, essa barreira sairá em velocidade neutra, para trás da nave, ela dura apenas dois segundos após ter sido criada. Caso essa barreira atinja o outro player, ele sofre dos mesmos deméritos da habilidade acima.

Lembrando que, para a real funcionalidade dessa habilidade, é necessário que o player inimigo está muito próximo da outra nave que a criou, logo, ela é por si só balanceada para que seja usada em apenas certos momentos de estratégia.

Tem mais uma habilidade no jogo, que não consiste em atrapalhar, chamada de “Burst” no jogo, que se resume a um aumento instantâneo na velocidade atual da nave que a optou por utilizar, podendo assim criar uma enorme vantagem para o player que a conseguir obter.

Todas as habilidades são adquiridas aleatoriamente no decorrer do jogo.

### 3.2.6. Economia

O jogo não se utiliza de conceito e/ou mecânicas de economia.

### 3.2.7. Fluxo de Telas

O jogo consiste de três telas básicas no decorrer de gameplay, como citados brevemente no tópico “Resumo de Fluxo de Jogo”. As telas são as seguintes:

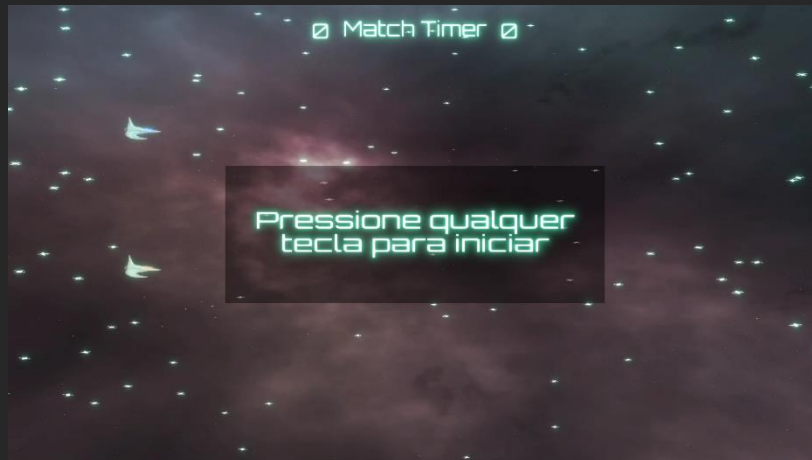
Tela de Menu Principal:

Ela consiste de apenas três botões, o primeiro que iniciará a corrida que se consiste o jogo, o botão de um breve tutorial e o botão para sair do jogo. Ao se utilizar do primeiro botão, vamos para a segunda tela do jogo.



Tela de Jogo:

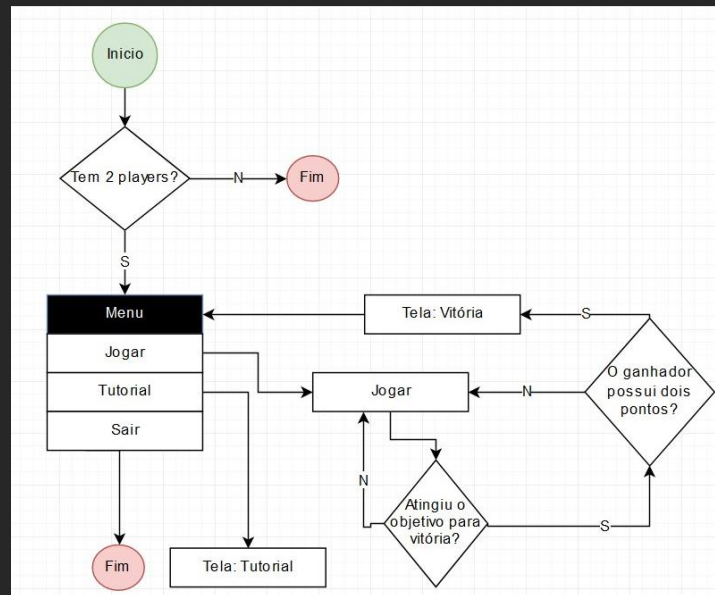
Onde todo o jogo e gameplay acontece, quando o objetivo do jogo for concluído, vamos para a terceira tela do jogo.



Tela de Vitória:

Uma simples tela que mostrará qual player saiu vitorioso da corrida, ela tem um timer de 5 (cinco) segundos obrigatórios, e logo após o seu fim, ao apertar de qualquer botão (seja no computador quanto no Arcade), o jogo voltará para a primeira tela do jogo, a de Menu Principal, para assim poder se iniciar uma nova partida, ou sair do jogo.





### 3.3. Opções do Jogo

As únicas opções do jogo se consiste a quando acelerar, a mudança de movimento para cima e para baixo da nave a partir da sua aceleração, e a utilização ou não das habilidades do jogo.

Para mais informações sobre a movimentação, checar o tópico “Ações”.

E para mais informações sobre as habilidades, checar o tópico “Combate”.

### 3.4. Rejogo e Armazenamento

O jogo não consiste de mecânica de Save Game.

Todo o sistema de rejogo se baseia em uma nova partida, se decorrendo do fato das partidas serem curtas, resultado do desejo do player caso ele queira uma nova partida ou não.

### 3.5. Trapaças e Descobertas

O jogo não possui um sistema de trapaças e/ou descobertas.

## 4. História, Configuração e Personagens

### 4.1. História e Narrativa

O jogo se passa em um futuro distante, onde a humanidade perdeu o seu senso de necessidade por entretenimento, mas que ainda existe um pequeno número de pessoas que procura diversão em algumas coisas.

Com o fácil acesso às tecnologias e combustíveis, um grupo de amigos começaram a praticar corridas espaciais na orbita de seu planeta, fazendo assim, competições saudáveis entre eles.

Além da sua história base de fundo, o jogo não consiste de narrativas.

### 4.2. Mundo do Jogo

### 4.2.1. Visão Geral do Mundo

O jogo se passa na terra, em um futuro distância, pouco se sabe sobre a terra em si, apenas sobre a órbita de seu planeta, que é aonde as corridas ocorrem.

### 4.2.2. Áreas

A área que se passa o jogo, é a órbita do planeta Terra, tudo que se pode ver é o vasto espaço sideral e algumas estrelas. Sua característica física se resume em apenas à gravidade nula do local, que já foi citado nos tópicos acima.

## 4.3. Personagens

O jogo tem apenas dois personagens, que sempre estão sempre dentro de suas naves, logo eles são diferenciados pelas suas cores, sendo:

- Laika – Vermelho



- Sputnik – Azul



## 5. Fases / Níveis

### 5.1. Fase Base

O jogo, como já citado, consiste de apenas uma fase, que é a corrida.

Todos os detalhes sobre ela podem ser lidos nos tópicos sobre “Jogabilidade e Mecânicas”.

### 5.2. Fase de Treinamento

O jogo não tem uma fase de treinamento.

## 6. Interface (imagens)

### 6.1. Sistema Visual

O jogo tem uma simples HUD, que consiste das seguintes informações:

Velocidade atual da nave.

Timer da partida.

Unidades necessárias para a vitória.

Distância em Unidades do outro player.

Timer ou nome da habilidade atual do player.

Todas as informações técnicas sobre esses itens podem ser encontradas no tópico “Fluxo de Jogo”.

A câmera se utiliza da perspectiva ortográfica, que segue sempre os players, dividindo a tela em dois em modo horizontal. Caso as naves estejam próximas uma da outra, a câmera será mudada para uma câmera única, ainda em perspectiva ortográfica, que focará nos dois players.



## 6.2. Sistema de Controle

As utilizações dos controles já foram citadas no tópico “Como se movimentar pelo jogo”, cabendo a este atual tópico apenas as descrições técnicas e visuais.

Toda a descrição visual, perante o Arcade, será apresentada no tópico “Sistema de Ajuda”, para evitar redundância deste documento.

## 6.3. Áudios, Músicas e Efeitos Sonoros.

O jogo tem áudios simples, que podem ser numerados das seguintes formas.

São duas músicas totais no jogo, que são:

A música do Menu Principal, “Hero Imortal”.

E a música durante a gameplay, “Fight Looped”.

As duas músicas citadas serão tocadas em loop até que algo a interrompa, como o fechamento do jogo, ou a mudança entre as telas.

O jogo também tem os seguintes efeitos sonoros:

Um simples efeito de botão, utilizado no Menu Principal, ao pressionar um botão.

Três sons para o timer inicial da partida, sendo um para cada segundo, outro para quando o timer iniciar a partida do jogo, e o último para caso o player queime a largada.

Três sons de efeito para cada habilidade existente em jogo.

E um simples som uniforme na tela de vitória do jogo.

Todos os áudios citados acima foram utilizados com as suas respectivas licenças, não violando nenhuma regra e/ou leis aplicadas a elas.

## 6.4. Sistema de Ajuda.

O jogo tem uma simples tela de ajuda, que explica rapidamente os botões utilizados no jogo sob a perspectiva do seu uso no Arcade. Como citado na imagem abaixo:



## 7. Oponentes

### 7.1. Inteligência de Oponente / Inimigo

O jogo é obrigatoriamente designado para dois players humanos, sendo assim, não existe uma mecânica de inteligência artificial para o mesmo.

### 7.2. Personagens Amigáveis e de Não-Combate.

O jogo não possui de personagem adicionais.

### 7.3. Inteligência de Suporte

O jogo tem um simples sistema de colisão invisível, localizado em um lugar fixo em cima e em baixo da onde a partida ocorre. Apesar do mapa, inicialmente, parecer infinito, esses colidores limitam a movimentação vertical dos players, impedindo que eles se distanciem muito neste eixo. Isso simplifica a criação de mapa, e a utilização de habilidades que dependem de um impacto físico.

## 8. Requisitos Técnicos

### 8.1. Hardware Alvo

O Hardware Alvo deste jogo foi o Arcade, logo, foi visado para poder rodar em qualquer sistema simples, para mais detalhes, pode-se citar os seguintes requisitos mínimos para o jogo.

Requisitos mínimos

Sistema Operacional: Windows 7 ou acima.

Processador 2Ghz e/ou acima.

Memória: 4Gb de RAM.

Placa de Vídeo: Não necessário.

Rede: Conexão com internet não necessária.  
Armazenamento: 200Mb de espaço disponível.

## **8.2. Hardware e Software de Desenvolvimento**

O jogo foi desenvolvido em um computador com os seguintes itens de Hardware:  
Sistema Operacional: Windows 10.  
Processador: Ryzen 5 2600.  
Memória: 16Gb de RAM.  
Placa de Vídeo: AMD Radeon RX580 8Gb.

O Software de desenvolvimento foi:  
Unity Engine.2019.

## **8.3. Requisitos de Rede**

O jogo não possui de requisitos de rede, sendo possível jogar completamente off-line.

## **9. Arte do Jogo**

Neste tópico, serão discutidos desde as artes iniciais do design do jogo, até como que elas chegaram em sua conclusão. Serão usados de rascunhos e pensamentos usados no processo de produção, com as suas seguintes explicações.

### **9.1. Design Inicial**

Inicialmente, a ideia do jogo foi para que fosse uma corrida em que você atrapalhasse os outros players como caixas e alguns poderes de alteração de mapa, logo, a ideia inicial foi que os personagens fossem literalmente humanos.

Posteriormente, a ideia foi mudada devida à algumas discussões, como a percepção de quanto trabalho teria apenas com o processo inicial, como a criação de sprites para todas as animações de corrida e colisão que iria ter no processo do jogo.

Rapidamente, a ideia foi mudada para naves em uma corrida espacial.

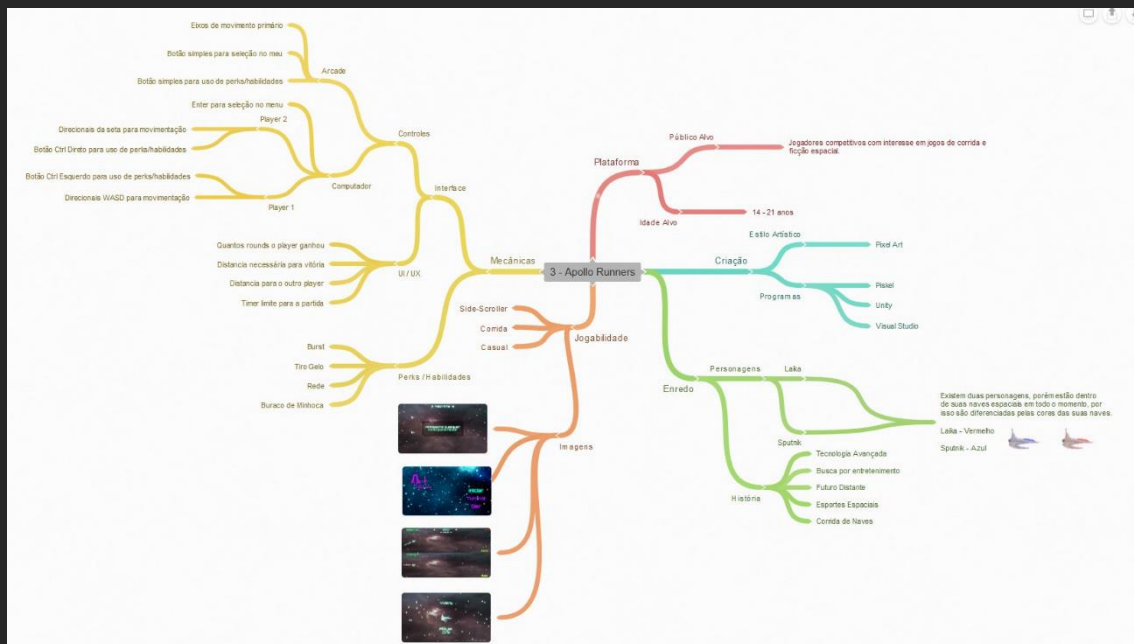
Pela ausência de terreno, como antes previsto, as habilidades para atrapalhar o outro player tiveram que ser mudadas também.

Segue abaixo alguns rascunhos sobre o desenvolvimento inicial:









[Link para o coogle.it sobre a imagem acima](#)

## 9.2. Sprites

Os sprites inicialmente foram designados para serem feitos em Pixel Art, os mesmos foram aplicados em alguns recursos 2D utilizados no jogo, como a nave e a logo da tela inicial:



Mas por algumas causalidades durante o processo de produção, alguns processos tiveram que ser adaptados para poder tornar a conclusão do projeto possível, como a imagem de fundo do Menu Principal.

Utilizada de uma foto real de uma representação do Espaço Sideral, a imagem foi deteriorada por meio de programas de edição para que não se destoasse muito do logo e pudesse manter uma congruência com o contexto.

O mesmo processo foi utilizado diversas vezes.

Um exemplo da imagem citada acima, pode ser vista na capa deste artigo.

Com o déficit de sprites, foi necessário a utilização de recursos como Partículas, que serão explicadas no próximo tópico.

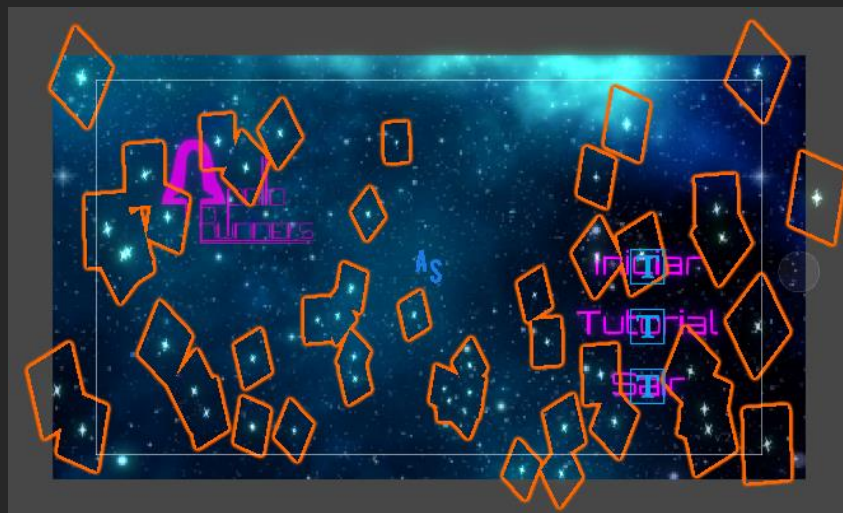
### 9.3. Partículas

Para poder prosseguir com a produção, sem a utilização de sprites, foram utilizados de partículas, do Particle System da Unity Engine. Sendo assim, foi possível que com apenas uma única imagem, e algumas linhas de programação, fosse produzido recursos adicionais e importantes para o jogo, simulando uma Pixel Art, e não se destoando do jogo.

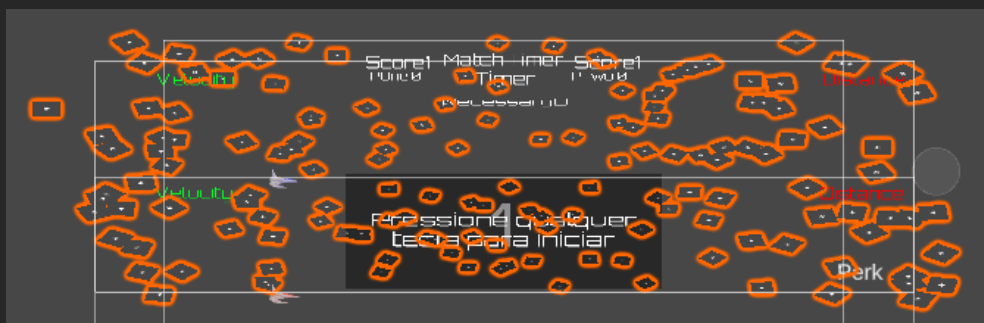
Tais recursos serão detalhados abaixo.

#### 9.3.1 Background

Para o menu não ficar estático, foi utilizado de partículas em loop, tomando a tela inteira, para passar a sensação de pequenas estrelas atravessando a tela lentamente. Elas foram programadas para alterarem, sutilmente, o seu tamanho e a sua rotação durante a sua trajetória.



O mesmo se aplica para a tela da corrida, e a tela de vitória.

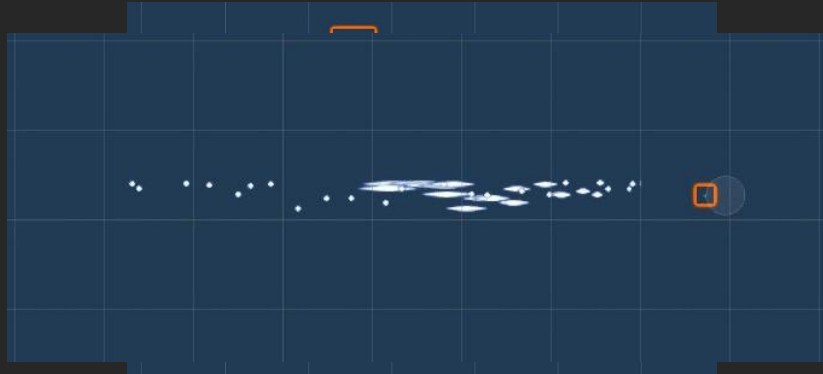


Com a utilização de um Background, se mostrou necessário uma dedicação maior para esse tópico, para assim existir uma sensação de movimento para o player, foi utilizado também de Skybox, que será devidamente explicado no próximo tópico, “Skybox”.

#### 9.3.2. Habilidades

Todas as habilidades utilizadas no jogo foram criadas a partir do sistema de partículas citada acima.

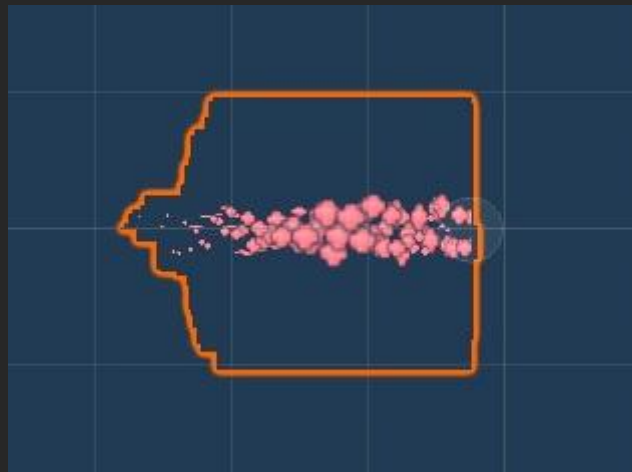
Segue abaixo alguns exemplos visuais de como elas são, visualmente, no processo de sua produção.



### 9.3.3. Propulsores

O sistema de partículas também foi utilizado nos propulsores da nave, a partir de linhas de programação, foi estipulado a sua força e quando ele seria utilizado ou não a partir da aceleração da nave alvo.

Mais detalhes serão descritos no tópico “Programação”.

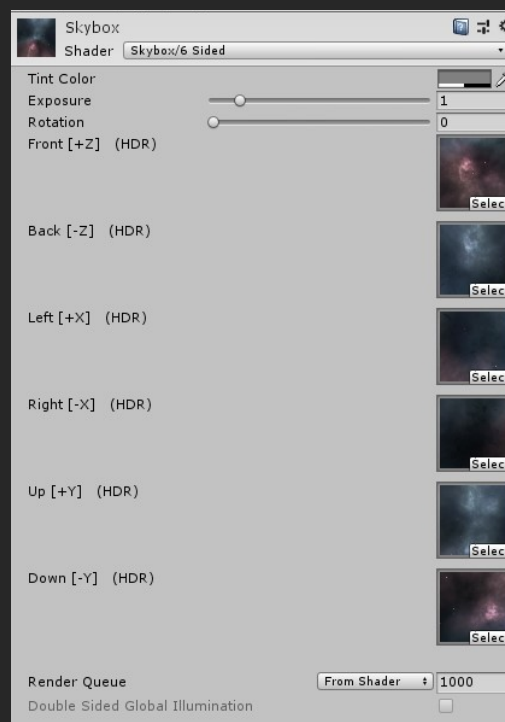


## 9.4. Skybox

Skybox é um sistema de seis imagens que simula um Background por volta de todo o mapeamento da cena, como se fosse uma enorme caixa englobando tudo, por isto o seu nome, Skybox.

Seguindo com isto, por o jogo ser, teoricamente, infinito no seu eixo horizontal, foi escolhido o uso de Skybox por sua facilidade e a otimização final que ele oferece.

Todas as imagens para o mesmo foram otimizadas para serem usadas em um sistema simples, que não precisasse de muito processamento, já que isso era visado para o produto final.



Esse sistema de Skybox foi utilizado tanto na tela da corrida, quanto na tela de vitória.

## 9.5. Programação

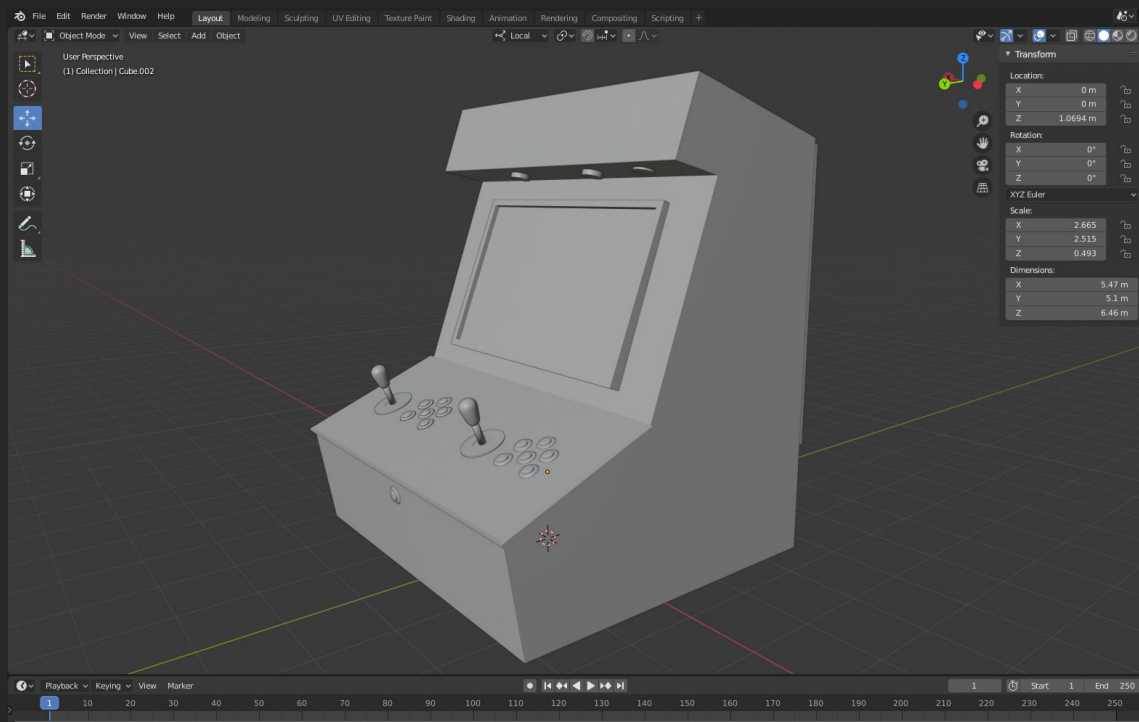
O jogo foi inicialmente designado para ser procedural, e com ferramentas de Sandbox, sendo assim, todas as suas linhas de programação foram feitas para serem facilmente adaptadas para toda e qualquer situação, dependendo da escolha do player.

Assim, com as suas 1848 linhas de código, qualquer usuário, futuramente, será capaz de editar, adaptar, ou adicionar qualquer conteúdo facialmente pelo o seu sistema interno procedural. Como, novas naves, habilidade e mapas.

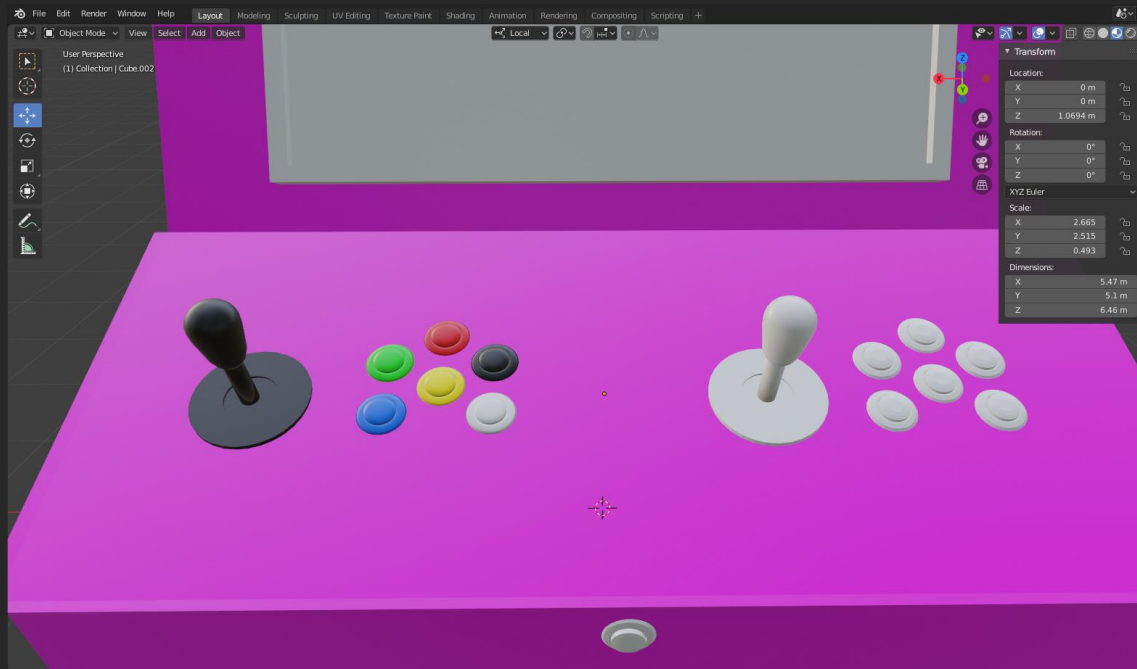
Qualquer detalhe adicional, assim como o próprio código base do projeto, pode ser encontrado em sua página do GitHub.

## 9.6. Recursos Adicionais

Para facilitar alguns processos durante a produção, alguns recursos visuais foram criados a partir do auxílio de ferramentas e objetos 3D, o objeto citado é uma recriação fiel do Arcade da faculdade PUC, localizada em Campinas, São Paulo, Brasil.



Seguindo esse modelo, com uma simples edição de texturas e shaders, as cores dos botões também foram simuladas, para poder ser utilizados na tela de tutorial, como mostrado no tópico “Sistema de Ajuda”.



## 9.7. Pós Processamento

Como todo projeto, a última etapa é configurar o sistema de pós processamento para esculpir o resultado final do jogo.



Para esse projeto, foi usado o efeito de Motion Blur durante as partidas, para dar a sensação de movimento das naves, e também o efeito de Blur de iluminação, para poder criar um contexto mais “sci-fi” para a obra.





O efeito de Motion Blur foi programado para aumentar, progressivamente, dependendo da velocidade em que a nave se encontra.

## 10. Game Design Canvas

**Apollo Runners**

<p><b>Plataforma</b></p> <p>Computador (Win) Arcade Público Joven / Adulto (+14) Data de lançamento prevista: 26 de Novembro de 2019</p>	<p><b>Conceito</b></p> <p>Num futuro distante, onde o entretenimento se tornou raro, um grupo de pessoas procuram diversão em corridas espaciais, onde o objetivo deles é atingir uma certa distância do outro, enquanto o atrapalha a fazer o mesmo</p>	<p><b>Jogabilidade</b></p> <p>Corrida espacial com a utilização de algumas habilidades para auxiliar a vitória, ou atrapalhar a vitória do adversário, até atingir a distância do objetivo.</p>	<p><b>Fluxo do Jogo</b></p> <p>A cada 5 segundos, cada player recebe uma habilidade aleatoriamente, podendo assim optar por usar ou não a habilidade, e onde quiser, abrindo assim a possibilidade de criação de diversas estratégias.</p>	<p><b>Controle</b></p> <p>Controle básico de movimento da nave, sendo Computador ou Arcade, e dois botões adicionais para resetar a rotação da nave e a utilização das habilidades.</p>
<p><b>Interface</b></p> <p>Teclado ou Joystick/Arcade Com músicas temáticas e efeitos em Pixel Art</p>	<p><b>Mundo do Jogo</b></p> <p>Se passa na terra, em um futuro distante, perto de sua órbita, em uma corrida espacial.</p>	<p><b>Chefes e Inimigos</b></p> <p>O único inimigo do jogo, é o player adversário, que terá a mesma possibilidade do que você para obtenção e uso de habilidades, também com a mesma nave e valores de velocidade. Cabe a eles a sua estratégia.</p>	<p><b>Mecânicas</b></p> <p>As mecânicas cabem às habilidades utilizadas em jogo, que são três: Uma barreira para impedir ultrapassagem, um burst que acelera a velocidade da nave e um tiro que atravessa o mapa e congela o inimigo.</p>	<p><b>Personagens</b></p> <p>O jogo consiste de dois personagens, que se alteram apenas na cor de suas naves, sendo Laika para a nave vermelha, e Sputnik para a nave azul.</p>

**As Abismo Studios**

Baixar GDD

**Plataforma**  
Computador (Win)  
Arcade  
Publico Joven / Adulto (+14)

### **Conceito**

Num futuro distante, onde o entretenimento se tornou raro, um grupo de pessoas procuram diversão em corridas espaciais, onde o objetivo deles é atingir uma certa distância do outro, enquanto o atrapalha a fazer o mesmo.

### **Jogabilidade**

Corrida espacial com a utilização de algumas habilidades para auxiliar a vitória, ou atrapalhar a vitória do adversário, até atingir a distância do objetivo.

### **Fluxo do Jogo**

A cada 5 segundos, cada player recebe uma habilidade aleatoriamente, podendo assim optar por usar ou não a habilidade, e onde quiser, abrindo assim a possibilidade de criação de diversas estratégias.

### **Controle**

Controle básico de movimento da nave, sendo Computador ou Arcade, e dois botões adicionais para resetar a rotação da nave e a utilização das habilidades.

### **Interface**

Teclado ou Joystick/Arcade

Com músicas temáticas e efeitos em Pixel Art

### **Mundo do Jogo**

Se passa na terra, em um futuro distante, perto de sua orbita, em uma corrida espacial.

### **Chefes e Inimigos**

O único inimigo do jogo, é o player adversário, que terá a mesma possibilidade do que você para obtenção e uso de habilidades, também com a mesma nave e valores de velocidade. Cabe a eles a sua estratégia.

### **Mecânicas**

As mecânicas cabem às habilidades utilizadas em jogo, que são três:

Uma barreira para impedir ultrapassagem, um burst que acelera a velocidade da nave e um tiro que atravessa o mapa e congela o inimigo.

### **Personagem**

O jogo consiste de dois personagens, que se alteram apenas na cor de suas naves, sendo Laika para a nave vermelha, e Sputnik para a nave azul.