**Projeto Zero**

Sprint Sexta 0800 => 1230 Alfa do Zero

Documento para listar coisas a serem feitas no Zero

Não apaguem o que já foi feito, apenas sinalizem o estado em que está.

**O QUE ESTIVER FEITO COLOQUEM [OK] NA FRENTE**

**SEMPRE LEIAM O GAME DESIGN!**

Caso alguém queira procurar efeitos sonoros, nesse site tem um monte com licença livre:

<http://soundbible.com/royalty-free-sounds-1.html>

**Símbolos:**

**P** Problema

**G** Gráfico

**E** Extra

**I** Interface

**S** Som

**V** Valor (variável)

**GD** Game Design

**Prioridade:**

*A* Alta, imediata

*M* Moderada, mais rápido o possível

*B* Baixa, Ao final das anteriores.

*D* Prioridade irrelevante, desnecessário fazer quando der e se for interessante.

**Fazer:**

Telas:

Tela inícial

Tela Jogo [*A*]

Tela de Opções (pop-up)[*M*]

Opções:[*B*]

Tirar Som

Tirar Musica

Resetar jogo

Sair

Som:[*M*]

musica de fundo

efeito de adição[OK]

efeito de subtração[OK]

efeito de multiplicação[OK]

Botões [OK]

Arte:

Botões[*A*] [OK]

Icone[*OK]*

(o resto ta pronto mas tem de vir para o drive)

Substituir botões por blocos [*M*]

Level Design

Mensagens enviadas ao longo do jogo[*B*][Não implementado]

Achievements[*A*][não implementado]

Scripts Necessários:

Tempo de spawn para os Blocos[*A*](OK)

Blocos de |1| a |9| [*A*](OK)

Liberar o valor quando jogador pontuar mais que 2 elevado ao valor(OK)

Implementar Achievements[*M*]

Implementar Missões[*M*]

**Desenvolvimento**

-Blocos

-Tipo: Os blocos do jogo são do tipo Button, que tem várias funções prontas para interação.

-Funcionalidade: Gerar o bloco com um valor aleatório ao inicializar.

\*Foram criadas três funções, uma para gerar o tipo do botão (soma, subtração e multiplicação), outra para gerar um valor aleatório, baseado no tipo, e uma para pegar a cor, em relação ao tipo.

\*A cor é alterada no componente Image do objeto, e o valor é atribuido ao componente Text no objeto filho deste.

-Funcionalidade: Mostrar o valor zero, após juntar valores opostos (Zerar)

\*É mudado o tipo, o valor e a cor do objeto, e é acionado a opção de desvanecer, a qual diminui o componente Alpha da cor da imagem do objeto, e após um tempo predeterminado, o objeto é destruído.

-Funcionalidade: Escolher objeto

\*Ao pressionar o botão do mouse (ou toque) sobre o objeto por mais de um frame (mais ou menos 60 ms), ele é considerado como sendo carregado. Essa opção é chamada no evento Initialize Potential Drag, do objeto Button.

\*Quando isso acontece, o pai do objeto é alterado para "null" (sem pai), e depois atribuido de volta ao pai original. Ao fazer isso, o objeto é elevado na hierarquia, e aparece por cima dos outros.

-Funcionalidade: Mover o blocos

\*Apenas atribui a posição do objeto à posição do mouse (ou toque).

\*Essa função é acionada no evento Drag do objeto Button, que é ativada quando o mouse (ou toque) está pressionado, e sua posição é alterada.

-Funcionalidade: Soltar blocos

\*Essa funcionalidade é explicada no gerenciador de blocos.

-Funcionalidade: Atualização

\*É verificado se o marcador de desvanecer (atributo "sumir") é verdadeiro. Se for ele muda o Alpha da cor da imagem do objeto, para que fique cada vez mais transparente, e após esse valor ficar negativo, o objeto é destruido. OBS: O tempo para o Alpha chegar a zero é atribuido no atributo "tempoSumir" do script GerBotao.

-Gerenciador de Blocos

-Tipo: Um objeto do tipo Panel, que é o pai dos objetos do jogo, e gerenciador dos mesmos.

-Gerenciamento dos blocos pelo cenário

\*O gerenciamento dos blocos é divido por algumas listas

\*Lista "objetos": Ela contem todos os blocos adicionados ao cenário, para manter um controle e poder reinicializar, juntar ou rearranjar eles mais facilmente.

\*Lista "grade": Um vetor com todas as posições possíveis para os objetos, para não precisarem serem feitos cálculos de posições, e deixar a quantidade de posicionamento mais genérica, e não depender de código para ajeitar a grade.

\*Lista "posicoesLivres": lista que contem todas as posicoes que podem ser adicinados novos objetos, ou movidos os existentes. Se a quantidade de itens nela for igual ao máximo permitido, não é possível adicionar novos blocos ao cenário.

\*Lista "posicoesOcupadas": o oposto da lista de posicoes livres. Existem as duas listas para facilitar o gerenciamento de posições, para não ser necessário percorrer a lista vazia todas as vezes para ver se existe um local ocupado.

-OBS: Muitos métodos desta classe são estáticos. O método "Soltar()" do gerenciador de blocos chamada o método "SoltarObjeto(GerBotao)" do gerenciador do jogo de forma estática, para não ser necessário passar a referência da instância desta classe para cada bloco. Detro de funções estáticas só podem haver chamadas para outras funções ou variáveis estáticas.

-Funcionalidade: Inicialização

\*Prepara o cenário, ajeitando as variáveis dinâmicas, ajeitando a grade de objetos, e adicionando os blocos iniciais a ela.

-Funcionalidade: Soltar um bloco

\*Quando um bloco é solto em qualquer parte, essa funcionalidade é chamada. Ela verfica se o bloco está na distância mínima para ser juntado com algum outro bloco.

\*Se estiver próximo o suficiente, ele junta ambos os blocos, chamando a funcionalidade "Zerar" de um dos blocos, e destrói o outro, além de criar um novo bloco em uma posição aleatória. Mais detalhes na funcionalidade "Juntar Blocos"

\*Se não estiver próximo o suficiente de nenhum outro bloco, é buscada a posição livre mais próxima deste bloco, para recolocá-lo na grade.

-Funcionalidade: Juntar Blocos

\*São verificados os tipos dos blocos que serão unidos.

\*Se nenhum deles for multiplicador, os valores simplesmente são somados.

\*Se ambos forem multiplicadores, o valor deles é multiplicado, e o tipo da junção continua sendo multiplicador.

\*Se apenas um for multiplicador, os valores são multiplicados, e o tipo passa a ser o do qual não é multipicador (soma ou subtração)

\*Se o valor resultante for diferente de zero, um dos objetos é destruído, e ao outro é atribuído este valor.

\*Se o valor for zero, um dos objetos é destruído, o outro é zerado, e o jogador recebe pontos iguais ao valor positivo da junção (ex: +2 com -2 resulta em 2 pontos). Também é adicionado um novo objeto ao cenário, quando o valor é zerado.

\*É verificado a pontuação, para aumentar os valores máximos dos blocos. <IMPLEMENTAR>

-Funcionalidade: Reiniciar cenário

\*Destrói todos os objetos existentes no cenário e limpa as listas de posições ocupadas e posições vagas, depois adiciona todas as opções do cenário como vagas livres.

-Funcionalidade: Adicionar um novo Bloco

\*Para fins de testes, é possível adicionar blocos com valores não aleatórios ao cenário.

\*É buscado uma posição vaga aleatória na grade do cenário.

\*Se existir uma posição livre, é criado um novo objeto de bloco, que recebe um tipo aleatório, sendo que a chance de sair positivo ou negativo é igual, e a chance de sair multiplicador é muito inferior.

\*O objeto é adicionado à grade, e a lista de objetos. A posição que ele ocupa é passada da lista de vazios para a lista de ocupados.

\*Se não existir um local livre, é mostrada uma mensagem dizendo que o cenário está cheio. <IMPLEMENTAR>