**O jogo**

Galactic Crusade é um jogo de estratégia, top-down, turn-based onde o jogador controla um robô e deve sobreviver ao maior número de níveis que conseguir. A temática do jogo é espacial, cada inimigos eh bem diferente e possui uma habilidade específica para dificultar a jornada de quem está jogando. O principal diferencial buscado é o de trazer uma experiência que se adapta ao jogador atual, analisando seus padrões, comportamentos e perfil de ataque/movimentação para fazer com que ele se sinta sempre desafiado enquanto se diverte.

**Inspirações**

A ideia inicial do projeto partiu do objetivo de criar um jogo baseado principalmente em elementos de estratégia e que ao mesmo tempo trouxesse sempre a sensação de uma experiência única para cada jogador. Partindo disso, foram pensados e pesquisados vários jogos que trazem propostas semelhantes como base e inspiração para os elementos desenvolvidos: Crypt of necrodancer, sproggiwood, hoplite, hitman Go, etc.

Cada um deles possui uma temática bem única, tanto como jogabilidade, arte e experiências proporcionadas. Considerando isso, busquei entender um pouco das mecânicas aplicadas por eles para trazer algo semelhante no meu projeto.

**Desenvolvimento**

Desde o começo, o objetivo foi de sempre buscar primeiramente as mecânicas fundamentais do jogo, visando possibilitar o teste de cada elemento desde sua fase inicial. Tendo isso em mente, o processo se iniciou com a montagem do gerador de níveis, para que já fosse possível ver o grid visualmente e, consequentemente, possibilitasse e criação do personagem principal e sua movimentação por turnos.

Após a movimentação do jogador, o pathfinding com A\* dos inimigos foi implementado. Com isso, foram realizados vários testes incrementais para sempre buscar a melhor maneira de se desenvolver estes elementos fundamentais, com foco na movimentação do jogador, para que aconteça de uma forma que faça sentido e seja fácil/intuitiva para quem quer que esteja jogando.

A terceira fase do projeto foi a mais extensa, onde foram pensados e trazidos novos elementos para o gameplay como um todo. Primeiramente, veio o desenvolvimento dos poderes do jogador para que cada um tivesse uma certa estratégia envolvida, juntamente com um balanceamento inicial. Para aumentar as possibilidades e dificuldades nas fases, foram também desenvolvidas armadilhas de cenário, que ao contrário dos inimigos não atacam, mas possuem efeitos que visam fazer com que o jogador tenha que planejar melhor por onde anda. Juntamente com isso, nesta parte comecei a fazer os modelos 3D, que serão melhor explicados a seguir, e também habilidades secundárias para cada um dos inimigos para que sejam diferentes um dos outros.

Por último, foi implementado o leitor de níveis, para que possibilite a utilização de níveis pré-montados, além dos procedurais. A ideia é que os 10 primeiros níveis não serão procedurais, para coletar informações sobre padrões e comportamentos do jogador primeiramente e então criar níveis com inimigos e obstáculos que façam mais sentido e tragam maiores desafios e diversão, a partir desta informação coletada. Além disso, foram os menus iniciais foram criados, som adicionado e vários bugs foram corrigidos.

**Estrutura**

O projeto em si baseia-se principalmente nos scripts ‘managers’:

Um que controla o campo de batalha (matriz) e todos objetos que se movimentam nele, outro responsável pela progressão de níveis, um terceiro que faz o controle de todos os elementos da interface e por último o gerenciador de inimigos, que utiliza a técnica MiniMax para montar suas jogadas.

BoardManager - controla a matriz de todos os objetos que compõem cada nível, bem como todas as validações para movimentação de cada personagem, etc.

GameManager – responsável por controlar o carregamento de novas fases e limite de FPS

UXManager – controla todos os botões e as chamadas feitas pelos mesmos

EnemyCoordinator – Controle de quando cada um dos inimigos age na partida

Além disso, para realizar as trocas de turno entre jogador e inimigos, foi implementado o sistema de eventos da Unity (conforme tutorial XXXXXXX).

Sobre a estrutura das classes, há duas principais das quais o jogador e inimigos fazem uso (através de polimorfismo): Character, que é a classe que permite a movimentação do objeto na matriz e Entity, que que possui as informações de pontos de vidas do objeto. Já os obstáculos, utilizam-se apenas da classe Entity. Ou seja, todos os Characters são Entities, mas Entities não necessariamente são Characters. Fora isso, outra classe bastante utilizada foi a SpecialTile, que serve para situar elementos no cenário, fazendo com que cada um saiba sua posição na matriz, caso esta informação seja necessária durante o jogo.

**Arte**

Visto que arte e modelagem não são áreas com qual mais me identifico (ou tenho experiencias previas), um dos maiores desafios deste projeto foi pensar na arte e identidade visual que eu gostaria de trazer para o jogo. Depois de pesquisar várias opções, como assets e modelos open-source, etc. foi então que encontrei a ferramenta MagicaVoxel, que permite modelagem 3D de voxels. Devido a sua fácil curva de aprendizagem, juntamente com o fato de que a própria arte baseada em voxels pode ser considerada mais prática para alcançar resultados rápidos, decidi ir por este caminho e nele encontrei o modo do qual quero que meu jogo seja visualmente.

Além do fato desse estilo de arte estar em alta ultimamente, o fato de todos os modelos 3D do jogo serem modelados por mim mesmo auxilia para manter a coerência entre todos os elementos visuais.

**Estado atual**

Durante o semestre, foi possível desenvolver toda a fundamentação do jogo e elementos essenciais para que a versão atual seja completa no sentido da jogabilidade, com um início, meio e fim. No momento, o jogo está completamente funcional para a plataforma Android e possui os seguintes elementos:

- Gerador de níveis procedural e pre-configurados

- 10 niveis pre-montados para introduzir os elementos do jogo e capturar informações do jogador

- Personagem principal com 3 poderes, ataque básico e movimentação estendida/configurável

- 3 inimigos, onde cada um possui uma característica/habilidade secundaria única

- 2 obstaculos estáticos

- Menus

- Sons

- Efeitos especiais

O jogador pode escolher, além da movimentação e ataque básico, 3 poderes bem únicos que servem de auxílio para que consiga passar de fase em fase:

1 – Minas Remotas: Poderosas minas indetectáveis que possuem um tempo de ativação e causam dano em toda área ao seu redor.

2 – Overcharge (Sobrecarga): <botbot> usa as todas suas energias para liberar uma descarga elétrica em uma grande área, causando um alto dano a todos inimigos atingidos. Porém, esse poder drena toda sua bateria e o jogador fica 3 rodadas sem jogar

3 – Míssil teleguiado: Míssil XXX capaz de matar qualquer inimigo. O jogador deve selecionar o ponto de impacto, que será atingido após XX rodadas. Este poder só pode ser utilizado uma vez por nível.

Além disso, foram desenvolvidos 3 inimigos:

1 – Rockman: Poderoso inimigo feito de pedra com alta resistência e poder de ataque. Seus punhos são tão fortes que possuem a possibilidade de nocautear o jogador por 1 rodada;

2 – Veno-alien: Mesmo não sendo tão letal na luta corpo-a-corpo, este inimigo deixa sempre um rastro de veneno capaz de causar um alto dano ao jogador

3 – Lavamonster: Inimigo altamente perigoso por ser capaz de atacar de distancias maiores.