

AXOLOTL (nome provisório)

Resumo

AXOLOTL (nome provisório) é um jogo de aventura e plataforma em 2D, em que um pequeno robô deve explorar a fábrica submarina na qual ficou preso, derrotando diferentes criaturas e conseguindo assim os recursos para escapar desse labirinto subaquático. O diferencial do jogo consiste nas criaturas que interagem com o jogador, que são proceduralmente animadas e estão constantemente reagindo a tudo ao seu redor.

Inspiração e Motivação

A principal inspiração por trás de Axolotl é o jogo [Rain World](#), lançado pela Videocult em 2017. O jogo é primariamente centrado na sobrevivência e interação com criaturas com inteligência artificial (IA) complexa e dependente de parâmetros aleatorizados, gerando experiências de jogo completamente diferentes, mesmo com o cenário e a jogabilidade se mantendo constantes.

Isso nos motivou a explorar a ideia de um jogo de plataforma que, diferente da maioria, não utiliza criaturas com caminhos ou IAs fixos, que sempre as levam ao jogador, por exemplo. Entretanto, Rain World é mais centrado em fugir e tratar as criaturas como “obstáculos inteligentes”, enquanto a ideia de Axolotl é permitir uma interação maior e mais próxima com elas, por meio do combate.

A Fábrica Alagada

Todo o jogo se passa dentro de uma antiga fábrica de Sustância, um material radioativo peculiarmente saboroso que serve como combustível para os mais diversos tipos de maquinários. Depois que a plataforma que a sustentava desmoronou, poucos setores da fábrica continuam operando debaixo dos mares - e consequentemente atraindo a fauna local para a Sustância.

A Fábrica Alagada se divide em diversos quartos, que sempre se conectam a um ou mais quartos por meio de **esteiras**. As criaturas e o jogador podem se mover livremente por cada quarto e dele se locomoverem para um quarto adjacente.

Sustância

A sustância é o material comestível e combustível que é produzido pela Fábrica. Ela pode ser encontrada em diversas formas e lugares, sendo transportada por canos entre diversos quartos diferentes. Cada criatura percebe a sustância de forma diferente, seja a vendo ou sentindo seu cheiro.

Centro de Sustância

O Centro de Sustância é um quarto da Fábrica Alagada que ainda funciona e está constantemente produzindo sustância, servindo como porto seguro para o jogador. É um quarto com um grande galão de sustância que pode ser utilizado ilimitadamente para repor o estoque do jogador.

Quando o jogador usa o galão de sustância para preencher seu combustível, o jogo é salvo e o Centro de Sustância é definido como seu *respawn point*. Além disso, nascem novas criaturas na fábrica, de acordo com a **política de respawn**. Após selecionar quais criaturas serão geradas, o estado da tabela de criaturas é salvo para caso o jogador precise carregá-lo novamente. Isso significa que as criaturas que se moveram e estavam em outros quartos quando o jogo foi salvo serão carregadas nesses mesmos quartos.

Salas de Distribuição

As Salas de Distribuição se localizam ao redor do do Centro e possuem Distribuidores, que servem para retirar sustância dos canos que a carregam através da fábrica. Conseqüentemente, esses quartos são ocupados por criaturas mais fortes e territoriais (**alfas**), que defendem sua fonte de alimento fácil.

Escapando da Fábrica

No início do jogo, uma das primeiras salas que o jogador passará por será um grande portão, visivelmente trancado e com vários indicadores luminosos ao redor. Conforme cada Centro de Sustância é utilizado pelo jogador, os indicadores se acenderão de acordo e, com um número deles suficiente aceso, a porta irá se abrir e o jogador poderá completar o jogo.

Como cada Centro será rodeado por **salas de distribuição**, o jogador terá que obrigatoriamente se encontrar com criaturas e interagir com elas, seja afugentando-as, fugindo delas ou as combatendo diretamente. Além disso, nem todo Centro precisará ser acessado para completar o jogo, permitindo que o jogador escolha caminhos diferentes para explorar a fábrica e criaturas diferentes para interagir.

Esteiras

As esteiras são utilizadas para permitir a passagem do jogador e de criaturas entre dois quartos. Visualmente, cada esteira sempre tem uma parte visível e uma parte ocultada pela parede do quarto atual, onde se localiza sua *loading zone* (zona de carregamento).

Seu funcionamento segue o seguinte esquema:

- Se a esteira está parada, ela age como chão normal e não tem loading zone.
- Se a esteira está parada, ela pode receber uma criatura/jogador na parte visível ou na parte oculta, o que acontecer antes.
- Caso alguma das partes esteja ocupada, a esteira passa a se mover.

Sandbox

Para testar as mecânicas e as primeiras implementações do jogo, será feito um *sandbox* de algumas salas que poderão ser reaproveitadas depois. As imagens de planejamento estão disponíveis em [sandbox](#).

O Jogador

O jogador controla um personagem robótico que funciona à base de substância e precisa absorvê-la de tempos em tempos para continuar explorando. A quantidade de substância presente no robô é indicada visualmente e possui um valor máximo possível, que será definido após *playtesting*. A ideia central é que seja possível obter substância em porções menores para poder explorar por mais tempo. Dado que ela pode ser retirada tanto de outras criaturas como de partes da fábrica, isso deve incentivar o jogador a descobrir novas áreas e tentar entender que seres facilitam sua sobrevivência.

Habilidades

- Andar;
- Correr;
- Absorver *substância*;
- Pular;
- *Wall slide*: cai devagar quando adjacente a paredes e segurando na direção delas;
- *Wall jump*: pular na direção oposta de paredes quando adjacente a elas. O *wall jump* empurra o jogador para longe da parede, então não é possível usar vários consecutivamente na mesma parede;
- Pendurar-se: apertando um botão direcional para cima quando estiver em uma *ledge* o jogador pode se pendurar nela. Enquanto pendurado, ele pode tanto pular na horizontal como em um *wall jump*, pular para cima ou se soltar;
- Interação com certos objetos:
 - Pegar
 - Jogar
 - Ativar (nem todos os objetos possuem uma ativação, caso o jogador tente ativar um deles isso será indicado por um *feedback* visual e/ou sonoro)
 - Soltar

As criaturas

A Fábrica Alagada é ocupada por diferentes criaturas, cada uma com suas características e comportamentos. Todas as criaturas estão de alguma forma relacionadas com a Sustância, seja como alimento ou alimento de seu alimento. Além disso, todas interagem com o jogador de alguma forma, seja ela pacífica, agressiva ou temerosa. Muitas delas possuem partes que podem ser usadas pelo jogador para navegar a fábrica ou combater outras criaturas.

A mente de uma criatura

Ainda existem muitos aspectos da IA de cada criatura para se pensar, sendo que vários devem sofrer alterações conforme aconteça o desenvolvimento e o *playtesting* do jogo. Porém, existem alguns modelos básicos que pretendemos trabalhar sobre. Isso também ajudará na parte de simulação para gerar consistência entre os comportamentos de cada tipo de criatura.

Estado atual

O estado atual representa o tipo de ação que a criatura está tomando no momento e é o principal componente para ditar seu comportamento. Algumas criaturas podem ter estados específicos, mas os estados mais simples e mais genéricos que temos em mente são os seguintes:

- *Idle*: a criatura está parada até receber um estímulo externo ou tomar uma decisão.
- *Wandering*: a criatura está se movendo sem propósito específico, apenas se deslocando por instinto.
- *Targeting*: a criatura encontrou outra criatura ou objeto - ou seja, um alvo - e está tomando uma decisão a respeito dele ou esperando um estímulo de seu alvo.
- *Chasing*: a criatura está ativamente se deslocando para alcançar um alvo e não tem nenhuma ação que possa fazer para interagir com ele, exceto persegui-lo.
- *Fighting*: a criatura está agindo para agredir outra criatura ou objeto.

- *Escaping*: A criatura está ativamente fugindo de outra ou de algum estímulo.

Ademais, todas as criaturas têm um estado especial quando estão mortas, em que não recebem estímulos, não podem tomar decisões e simplesmente aguardam outras ações do jogador ou da tabela de criaturas.

Estado mental

O estado mental é um conjunto de valores que representam o que a criatura está sentindo e são usados para influenciar suas ações de acordo. Cada elemento do estado mental é uma emoção que sofre alteração nos seus valores de acordo com parâmetros do tipo de criatura e de sua individualidade. Por ora, os valores definidos são:

- *Empty*: valor padrão que não gera nenhuma influência sobre os comportamentos da criatura.
- *Curious*: valor que faz a criatura querer investigar estímulos ou alvos.
- *Angry*: valor que faz a criatura querer atacar outras criaturas ou objetos.
- *Afraid*: valor que faz a criatura querer fugir ou perder interesse em algo.
- *Stable*: valor de “aprendizado” que faz com que a criatura negue influências de outros valores.

O estado mental está altamente relacionado com a capacidade de reação e percepção de uma criatura. Por exemplo, uma criatura que estava lutando contra outra e começa a fugir tem chance maior de tentar voltar a lutar se seu valor de *angry* estiver muito alto - ou de se distrair durante a fuga caso receba um estímulo e esteja com o valor de *curious* muito alto. Uma criatura que perde um conflito ou cai em um buraco sem querer pode acabar com um valor de *afraid* mais alto e evitar outros conflitos no futuro.

Individualidade

A individualidade é composta por pequenas variações geradas aleatoriamente para cada instância de criatura. Atributos como a chance de uma criatura tropeçar, errar uma mordida, ignorar o jogador, ficar agressiva com facilidade e *etc* são alguns dos exemplos de individualidade.

Criaturas simples

Nem todas as criaturas do jogo irão seguir as ideias e modelos anteriormente mencionados, principalmente porque seu comportamento será naturalmente mais simples - sem estados e com apenas um objetivo (e.g. fugir do jogador, perseguir o jogador, ir atrás de comida, apenas vagar pelo espaço, etc). Algumas criaturas também terão apenas uma função estética e portanto não precisarão de uma IA desenvolvida.

Pontos de geração (*Spawn home* e *Spawn point*)

Para cada criatura no jogo, existe um ponto em que ela será instanciada no mundo, conhecido como ponto de geração. Um ponto de geração pode ser utilizado por uma ou mais criaturas, que podem ou não ser de uma mesma espécie.

Podemos dizer o seguinte:

- Um spawn home é um ponto onde uma criatura é originalmente gerada e geralmente é acompanhado de indicações estéticas na tela. Ele também possui outras opções de criaturas para gerar caso a criatura original morra.
- Um spawn point é um ponto onde uma criatura pode ser instanciada quando o quarto passa a ser simulado, com base nas informações da tabela de criaturas.

Política de Respawn

Durante o jogo, as criaturas que forem mortas pelo jogador deixam de ser simuladas e dadas como vivas na tabela de criaturas. Elas só serão repostas pelo jogo quando o jogador utilizar um Centro de Sustância.

Quando o Centro de Sustância é utilizado, a tabela de criaturas varre todas suas entradas em busca de criaturas que não estão vivas e não estão em nenhum quarto atualmente simulado. Para cada uma delas, ela envia um comando para seu *respawn point*, que vai gerar uma criatura de acordo com suas regras internas.

Lista de criaturas

A seguir temos a descrição das criaturas planejadas para o jogo. Suas artes conceituais estão disponíveis na pasta [Criaturas](#).

Jelly

A criatura mais simples. Seu comportamento consiste em apenas boiar pela fábrica de forma aleatória, alternando entre impulsos em linha reta pelo espaço. Podem ser mortas pelo jogador, porém isso não gera benefícios.

Barracuda

O inimigo mais básico do jogo, um peixe carnívoro que ataca o que vê sem hesitar. Se a barracuda vê qualquer criatura, ela tem uma chance alta de parar o que está fazendo e ir atacar. Ela nada principalmente na horizontal, variando a inclinação às vezes.

Depois que a barracuda ataca algo, caso ela não tenha sucesso, há uma chance baixa de que ela desista e não ataque novamente.

Ouriço

O ouriço tem medo da maior parte das criaturas e foge quando se aproximam dentro do seu campo de visão. Seu casco é cheio de espinhos e caso outra criatura se aproxime demais, existe uma chance de que ele se lance contra ela.

Quando um ouriço morre, é possível usar seu casco como arma.

Camarão-pistola

O camarão-pistola é uma criatura passiva, andando aleatoriamente e perseguindo apenas ouriços quando entram em seu campo de visão. Ataca o jogador apenas quando se aproxima demasiado. Seu ataque consiste de um disparo de água utilizando sua garra traseira, que demora um certo tempo para ser armada.

Quando morto, é possível remover sua garra e utilizá-la como arma.

Simulação e IAs

Durante o jogo, sempre haverá pelo menos $N+1$ quartos carregados e sendo simulados como se o jogador estivesse neles, onde N é o número de quartos adjacentes ao quarto em que o jogador se encontra.

Nos quartos simulados as criaturas agem conforme suas IAs individuais, recebendo informações sensoriais como barulho e visão para que decidam como interagir com o terreno e outras criaturas presentes naquele quarto. Dessa forma, quando o jogador passar do quarto atual para um adjacente em que há uma ou mais criaturas elas já estarão agindo conforme seus comportamentos, dando uma sensação mais orgânica a essa transição.

Tabela de Criaturas

A Tabela de Criaturas serve para monitorar todas as criaturas do jogo e simular as criaturas que não estão em quartos carregados. Ela simula de forma simples as interações possíveis das criaturas com outras criaturas e o ambiente, como por exemplo aleatorizar a posição da criatura no quarto para simular movimento, fazer com que as criaturas se movam para quartos adjacentes baseados em chance, e fazer com que criaturas possam simplesmente morrer caso se encontrem em quartos com outras criaturas hostis sem precisar simular cada uma detalhadamente.

O intuito dessa tabela é limitar a quantidade de processamento necessária para a IA das criaturas e fazer com que o número de salas seja escalável independentemente da complexidade dessas IAs, já que a simulação da Tabela de Criaturas é bem mais rápida em comparação.

Camadas de simulação

Quando pensamos na simulação gráfica e física do jogo, principalmente considerando que nem toda ela será visível para o jogador, temos que pensar necessariamente nos recursos que serão utilizados para esse processo. Um excesso de recursos para simulações em *background* poderia prejudicar a experiência de jogo, gerando latência para o jogador.

Visto isso, os objetos de jogo terão sempre uma camada física e uma camada estética. Ambas as camadas estarão sujeitas às simulações de física, porém a camada estética só será considerada na simulação da tela atual. Desse modo, conseguiremos economizar recursos, pois será como se apenas um jogo visualmente mais simples estivesse rodando no background, porém com a complexidade de processamento necessária para gerar interações entre as criaturas como se estivessem na tela com o jogador.

Animação Procedural

Todas as criaturas serão animadas utilizando animação procedural, ou seja, simulando seus movimentos de acordo com regras pré-estabelecidas ao invés de exibindo sequências pré-construídas de *sprites*. Isso significa que podemos animar uma criatura com base em forças externas e comportamentos, além de aleatorizar pequenos elementos de cada animação, permitindo que cada interação com o jogador tenha simultaneamente previsibilidade e unicidade. O resultado esperado da utilização dessa prática é fazer com que o mundo pareça mais vivo e reativo à interação do jogador e que o comportamento das criaturas possa ser indicado através de feedback visual.

Ferramentas

Para o desenvolvimento do jogo, será usada a [Godot Engine](#).

Repositório no GitLab: <https://gitlab.com/lordanb/axolotl>.