

Jogos em Rede Multiusuário

Trabalho 2 - Escopo livre

Crie um jogo de escopo livre obedecendo às seguintes restrições:

Pre-game:

- O jogador deve ter uma conta no servidor, ou seja, ele deve poder se cadastrar com e-mail, senha e username.
 - Os usuários devem ser salvos em memória, arquivo ou em algum banco de dados.
 - É necessário verificar se o e-mail e o username são únicos.
 - O cadastro do jogador pode ser realizado em uma aplicação (ou página) separada.
- Implemente um sistema de "lobby", onde os jogadores logados, que não estão em uma partida em andamento, podem se ver e conversar em um chat global entre eles. Todas as mensagens no lobby são visíveis para os jogadores do lobby, mas não para os que estão em uma partida em andamento.
- Qualquer jogador pode "criar" uma **partida** que fica com o status "criada"
 - Partidas criadas, ficam visíveis no lobby e os outros jogadores podem "entrar" na partida.
 - Quando estiver com 2 ou mais jogadores na partida criada, o seu criador (somente ele) pode "iniciar" o game, o que fará com que a partida passe para o status "iniciada". Esse número mínimo de jogadores deve ser configurável em uma variável no código.
 - Os jogadores que não entraram na partida continuam no lobby.

Game-play:

- O jogo pode ser de qualquer gênero, porém as ações precisam ser em tempo-real, ou seja, não podem ser "em turnos".
- Qualquer tipo de mecânica pode ser implementada, entretanto todos os jogadores devem poder a qualquer momento realizar as ações. A não ser que o seu jogo tenha alguma mecânica que bloqueia as ações de um jogador dado algum evento.
- Deve haver algum controle de recurso único a cada jogador, por exemplo: Munições disponíveis, Itens em um inventário, Vidas, etc.
- Todos os recursos que cada jogador possui podem ser afetados (aumentados ou diminuídos) pelas ações durante o jogo.
- Cada jogador deve ter características que podem mudar durante o gameplay, por exemplo: velocidade, força, escudo, dano. Esses atributos podem ser alterados de acordo com as regras do seu jogo ou com as ações do jogador ou com os recursos que o mesmo tem no momento.
- Pelo menos 4 tipos de mecânicas devem ser implementadas, por exemplo: Movimento, aplicar dano, coletar itens, trocar de armas, entrar em locais de proteção, usar algum tipo de elemento na cena contra os adversários, etc.
- Não é necessariamente obrigatório o uso de websockets para a comunicação, o cliente e servidor podem ser implementados em qualquer linguagem, biblioteca ou motor de jogos.

Extras:

- Modo espectador: Onde um link(caso web) ou um código(caso seja um app) para a partida pode ser gerado para que alguém (mesmo não logado) possa assistir o gameplay.
- Otimização de dados: Qualquer otimização que reduza a quantidade de dados na comunicação entre servidor e cliente.

O trabalho é de escopo livre e serão avaliados os seguintes critérios com igual peso:

- Coerência: O jogo deve ter um objetivo claro e jogabilidade intuitiva
- Controles: O personagem deve ser controlado com um grau de dificuldade adequado para o estilo de jogo escolhido.
- Complexidade: Será avaliado o grau de complexidade do jogo desenvolvido.

Para entregar o projeto crie um link do github (ou similar) com o código do projeto e também um vídeo (disponível on-line) com a demonstração do jogo em ação. Envie os dois links pelo SIGAA, respeitando o prazo da tarefa, sempre com instruções de build e como jogar.