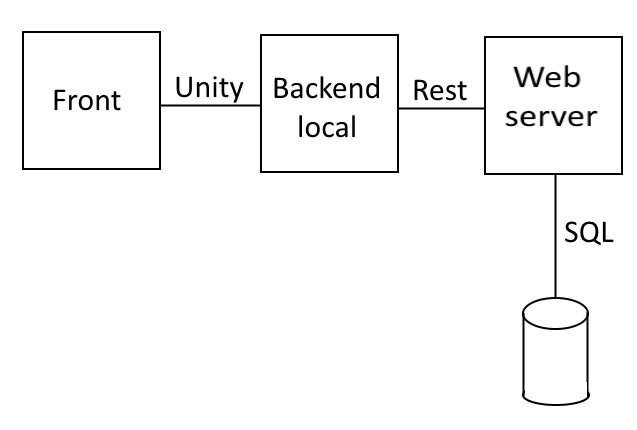
**Grupo: Jean Walper e Pedro de la Peña**

**Descrição do projeto**

O projeto consiste em uma simulação de itens, equipamentos e personagens em um jogo. Neste projeto é possível: criar uma conta com login, senha e apelido, criar até 3 personagens por conta, comprar itens de uma loja e também trocar itens entre usuários.

Os itens adquiridos podem ser equipados fazendo com que os atributos do personagem sejam melhorados. Além disso, os itens não são únicos: mais de um usuário pode possuir uma cópia do mesmo item ao mesmo tempo e, visto que os itens possuem raridades diferentes - e quanto mais raro mais difícil de conseguir e caro ele se torna - a troca entre os jogadores é um mecanismo ideal para aqueles que desejam atingir o seu pico de poder ou simplesmente gostam de colecionar itens. Contudo, não é possível que um mesmo usuário equipe dois de seus personagens com os mesmos itens: será preciso desequipar um para poder equipar outro.

**Diagrama da arquitetura do sistema**



**Dicionário Database**

**Tabela Users**

**---- Inputs do usuário ----**

Login: string única de 16 caracteres utilizada para entrar no jogo.

Senha: string de 64 caracteres utilizada em conjunto com o login para entrar no jogo.

Nick: string única de 16 caracteres que identificará os jogadores em jogo (apelido).

**---- Inputs do servidor ----**

Join\_date: data de criação da conta.

Last\_login\_date: data do último acesso do usuário.

Active: booleana (flag) que determina se o usuário logou recentemente.

Banned: booleana (flag) que identifica se as contas estão restringidas ou não de acessar o jogo.

Money\_spent: int que representa o dinheiro obtido pelo usuário em jogo.

**Tabela Item template**

**---- Inputs do servidor ----**

Template\_id: int única responsável para que o servidor identifique o item.

Item\_name: string que os players utilizarão para identificar o item no jogo.

Item\_type: string que os players utilizarão para identificar o tipo e utilidade do item no jogo.

Amount\_1 e Amout\_2: int que representa um campo genérico que pode ser utilizado em diversos tipos de itens para determinar seus efeitos. A partir destes campos é possível determinar a durabilidade de uma espada ou armadura e também a regeneração de vida de uma poção.

Rarity: int utilizada para determinar a raridade de um item; quanto mais alto o valor, mais raro o item será.

**Tabela Item**

**---- Inputs do servidor ----**

Item\_id: int auto incrementado que identifica um item quando este é adicionado a conta de um jogador.

Template\_id: int que faz referência à chave da tabela item\_template.

Login: string que faz referência ao login do usuário dono do item.

Equip: booleana que determina se o item está equipado ou não por um personagem.

Create\_date: data em que o item foi adicionado à conta.

**Tabela Character**

**---- Inputs do servidor ----**

Character\_id: int auto incrementado que identifica os personagens criados.

Login: string que faz referência ao dono do personagem.

Strenght: int que representa o atributo força do personagem.

Dexterity: int que representa o atributo destreza do personagem.

Vigor: int que representa o atributo vida do personagem.

Inteligence: int que representa o atributo inteligência do personagem.

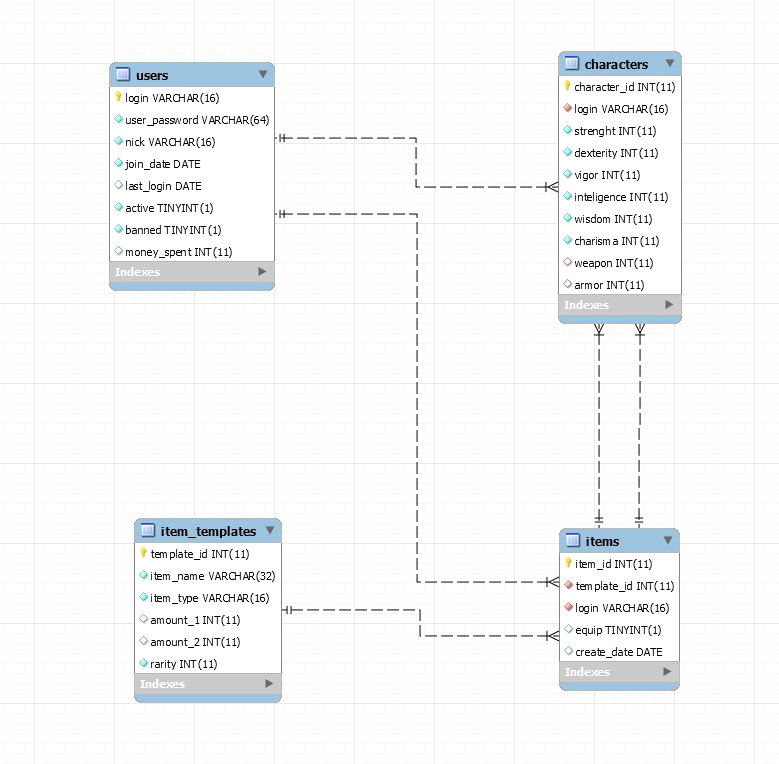
Wisdom: int que representa o atributo sabedoria do personagem.

Charisma: int que representa o atributo carisma do personagem.

Weapon: int que faz referência ao item utilizado como arma do personagem, aumentando seu ataque.

Armor: int que faz referência ao item utilizado como armadura do personagem, diminuindo a vida perdida quando o mesmo recebe dano.

**Diagrama E-R**



**REST API**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Recurso\Método | GET | POST | PUT | DELETE |
| /users | Retorna lista de todos os usuários  Return:  Success: Status 200, json array  Fail: Status 400, errormsg |  |  |  |
| /newuser |  | Adiciona um novo usuário, seguindo o modelo /newuser/:name/  :password/:nick/  :joindate/:lastlogin/:active/:ban/:money  Return:  Success: Status 200,  Json array  Fail: Status 400, errormsg |  |  |
| /user/:username | Retorna as informações do usuário especificado  Return:  Success: Status 200, json array  Fail: Status 400, errormsg |  |  |  |
| /user/:username/characters | Retorna os personagens criados pelo usuário especificado  Return:  Success: Status 200, json array  Fail: Status 400, errormsg |  |  |  |
| /newcharacter |  | Adiciona um novo personagem para o usuário, seguindo o modelo /newucharacter /:ownername/:str/:dex/:vig/:int/:wis/:cha  Return:  Success: Status 200,  json array  Fail: Status 400, errormsg |  |  |
| /putcharacter/ |  |  | Altera os atributos do personagem com base em seu id, , seguindo o modelo /putcharacter/:id/:strenght/:dexterity/:vigor/:inteligence/:wisdom/:charisma  Return:  Success: Status 200,  Json array  Fail: Status 400, errormsg |  |
| /userlogin/:username/:date |  | Altera a última data em que o jogador se conectou ao servidor (login mais recente)  Return:  Success: Status 200,  ok  Fail: Status 400, errormsg | O método originalmente era um patch/put, porém o Unity não estava aceitando tal método. |  |
| /deletecharacter/:id |  |  |  | Deleta um personagem com base em seu ID  Return:  Success: Status 200,  ok  Fail: Status 400, errormsg |
| /newitem/:template/:login/:date |  | Adiciona um novo item a um usuário e a data de sua criação.  Return:  Success: Status 200,  Json array  Fail: Status 400, errormsg |  |  |
| /deleteitem/:itemid |  |  |  | Deleta um item com base em seu ID  Return:  Success: Status 200,  ok  Fail: Status 400, errormsg |
| /item/:username | Retorna todos os itens possuídos por um usuário  Return:  Success: Status 200, json array  Fail: Status 400, errormsg |  |  |  |
| /item/:username/weapons | Retorna todos os itens do usuário que são do tipo “weapon”  Return:  Success: Status 200, json array  Fail: Status 400, errormsg |  |  |  |
| /item/:username/armor | Retorna todos os itens do usuário que são do tipo “armor”  Return:  Success: Status 200, json array  Fail: Status 400, errormsg |  |  |  |
| /templates | Retorna todos os templates de item já criados no jogo.  Return:  Success: Status 200, json array  Fail: Status 400, errormsg |  |  |  |
| /trade/start |  | Inicia uma transação  Return:  Success: Status 200  Fail: Status 400, errormsg |  |  |
| /trade/rollback |  | Realiza um rollback  Return:  Success: Status 200  Fail: Status 400, errormsg |  |  |
| /trade/commit |  | Realiza um commit  Return:  Success: Status 200  ok  Fail: Status 400, errormsg |  |  |
| /trade/:id/:login |  | Altera os itens de um usuário com base no id do item  Return:  Success: Status 200  ok  Fail: Status 400, errormsg |  |  |

**Bugs conhecidos**

Quando um usuário possui mais de um personagem e ele tem itens equipados neles, não há feedback em relação aos atributos dos itens equipados. Contudo, quando o jogador só tem um personagem, a mudança de equipamentos possui feedback instantâneo.