ASTI - FATEC

Astirina Group

**Projeto de**

**Banco de Dados**

*Strings Fighter*

Versão: 1.0

Integrantes do Grupo:

**Anderson Szalai 2080125-4**

**André Martins 2080126-2**

**Gabriel Roque 2080160-2**

**Leandro Monge 2080160-2**

Conteúdo

[1. Descrição do negócio 4](#_Toc264582926)

[2. Modelo conceitual 5](#_Toc264582927)

[2.1 Diagrama entidade-relacionamento 5](#_Toc264582928)

[3. Modelo lógico-conceitual 6](#_Toc264582929)

[3.1 Esquema de relações 6](#_Toc264582930)

[4. Normalização 7](#_Toc264582931)

[5. Dicionário de Dados Lógico e Conceitual 8](#_Toc264582932)

[5.1 Tabelas 8](#_Toc264582933)

[5.1.1 Tabela de Clãs 8](#_Toc264582934)

[5.1.2 Tabela de Tipo de Personagens 8](#_Toc264582935)

[5.1.3 Tabela de Jogadores 8](#_Toc264582936)

[5.1.4 Tabela de Personagens 9](#_Toc264582937)

[5.1.5 Tabela de Hospedagem das lutas 10](#_Toc264582938)

[5.2 Script de criação e população das tabelas 11](#_Toc264582939)

# Descrição do negócio

**JOGO STRINGS FIGHTER**. O projeto Strings Fighter consiste em um jogo de luta multijogador em rede para 2 pessoas.

Para gerenciamento das partidas, utiliza-se um web service atrelado a um banco de dados o qual armazena todos os clãs e personagens disponíveis para utilização pelos jogadores, além dos usuários jogadores, seus personagens cadastrados e informações sobre as salas de jogo criadas.

Os dados armazenados acerca dos usuários jogadores são utilizados para permitir o acesso ao jogo e para obtenção de estatística. Para cada personagem criado pelo jogador, são armazenados o nível e a quantidade de pontos de experiência obtidos ao longo das partidas.

As informações sobre as salas de jogo são utilizadas pelo jogo para realização de conexão direta entre jogadores via socket a partir de seus IPs/hostnames e para armazenamento de estatísticas sobre jogadores que venceram as partidas.

# Modelo conceitual

## Diagrama entidade-relacionamento

Pontuação

Senha

Ativo

**Código do tipo de personagem**

Título

Nome

**Código do jogador**

Hostname/IP desafiante

Código jogador vencedor

Hostname/IP criador da sala

Data de vitória

**Código da sala**

Experiência

Data de criação

Nível

**Codigo do personagem do jogador**

**Código do clã**

**Código jogador desafiante**

**Código jogador criador da sala**

Data de criação

Título

**1,N**

**1,N**

**0,N**

**1,N**

**1,1**

**1,2**

**1,1**

**1,1**

Participa

**Sala de jogo**

Possui

Possui

**Personagem de jogador**

Faz parte

**Clã**

**Tipo de personagem**

**Jogador**

# Modelo lógico-conceitual

## Esquema de relações

**tblcharactertype** (**idcharactertype**, idclan(fk), title)

**tblclan** (**idclan**, title)

**tblgameroom** (**idgameroom**, idplayercharacterowner(fk), hostowner, idplayercharacterchallenger(fk), hostchallenger, idplayercharacterwinner(fk), won\_at, created\_at)

**tblplayer** (**idplayer**, score, username, password, active)

**tblplayercharacter** (**idplayercharacter**, idplayer(fk), idcharactertype(fk), xp, created\_at)

# Normalização

**tblcharactertype**

1FN: PK definida; 2FN: PK simples; 3FN: não identificadas dependências transitivas

**tblclan**

1FN: PK definida; 2FN: PK simples; 3FN: não identificadas dependências transitivas

**tblgameroom**

1FN: PK definida; 2FN: PK simples; 3FN: não identificadas dependências transitivas

**tblplayer**

1FN: PK definida; 2FN: PK simples; 3FN: não identificadas dependências transitivas

**tblplayercharacter**

1FN: PK definida; 2FN: PK simples; 3FN: não identificadas dependências transitivas

# Dicionário de Dados Lógico e Conceitual

## Tabelas

### Tabela de Clãs

Esta tabela armazena os dados dos Clãs contidos nos jogo.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **tblclan** | (**idclan**, title) | | | | |
| **Nome Lógico** | | **Nome Conceitual** | **Tipo** | **Descrição** | **Obrigatório** |
| idclan  (Chave Primária) | | Código do Clã | Inteiro | Código do Clã. | Sim |
| title | | Título | Texto | Título do Clã. | Sim |

### Tabela de Tipo de Personagens

Esta tabela armazena os dados dos Tipos dos Personagens (ex: Letras, Símbolos ou Mestiços) contidos nos jogo.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **tblcharactertype** | (**idcharactertype**, idclan, title) | | | | |
| **Nome Lógico** | | **Nome Conceitual** | **Tipo** | **Descrição** | **Obrigatório** |
| idcharactertype  (Chave Primária) | | Código do tipo de personagem | Inteiro | Código do tipo do personagem. | Sim |
| idclan  (Chave Estrangeira referente a tblclan) | | Código do Clã | Inteiro | Código do Clã do qual o personagem faz parte. | Sim |
| title | | Título | Texto | Título do personagem. | Sim |

### Tabela de Jogadores

Esta tabela armazena os dados dos Jogadores cadastrados no jogo.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **tblplayer** | (**idplayer**, score, username, password, active) | | | | |
| **Nome Lógico** | | **Nome Conceitual** | **Tipo** | **Descrição** | **Obrigatório** |
| idplayer  (Chave Primária) | | Código do jogador | Inteiro | Código do jogador. | Sim |
| score | | Pontuação | Inteiro | Pontuação total do jogador obtida ao londo das partidas. | Sim |
| username | | Nome | Texto | Nome do usuário. É utilizado para acesso ao jogo. | Sim |
| password | | Senha | Texto | Senha de acesso do jogador. | Sim |
| active | | Ativo | Inteiro pequeno | Indica se o usuário está ativo e pode ter acesso ao jogo. | Sim |

### Tabela de Personagens

Esta tabela armazena os dados dos Personagens selecionáveis do jogo escolhido pelo jogador.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **tblplayercharacter** | (**idplayercharacter**, idplayer, idcharactertype, level, xp, created\_at) | | | | |
| **Nome Lógico** | | **Nome Conceitual** | **Tipo** | **Descrição** | **Obrigatório** |
| idplayercharacter | | Codigo do personagem do jogador | Inteiro | Codigo do personagem escolhido pelo jogador. | Sim |
| idplayer  (Chave Estrangeira referente a tblplayer) | | Código do jogador | Inteiro | Código do jogador. | Sim |
| idcharactertype  (Chave Estrangeira referente a tblcharactertype) | | Código do tipo do personagem | Inteiro | Código do tipo do personagem escolhido. | Sim |
| level | | Nível | Inteiro | Nível do personagem do jogador. Representa o quão evoluído o personagem está. | Sim |
| xp | | Experiência | Inteiro | Pontos de experiência do personagem do jogador. | Sim |
| created\_at | | Data de criação | Hora/data | Data de criação do personagem para o jogador. | Sim |

### Tabela de Hospedagem das lutas

Esta tabela armazena os dados do jogador que está criando a sala onde ele esperará por um desafiante para iniciar a luta.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **tblgameroom** | (**idgameroom**, idplayercharacterowner, hostowner, idplayercharacterchallenger, hostchallenger, idplayercharacterwinner, won\_at, created\_at) | | | | |
| **Nome Lógico** | | **Nome Conceitual** | **Tipo** | **Descrição** | **Obrigatório** |
| idgameroom | | Código da sala | Inteiro | Código da sala criada. | Sim |
| idplayercharacterowner  (Chave Estrangeira referente a tblplayercharacter) | | Código jogador criador da sala | Inteiro | Código do jogador que criou a sala. | Sim |
| hostowner | | Hostname/IP criador da sala | Texto | IP/hostname do jogador que criou a sala. | Sim |
| idplayercharacterchallenger (Chave Estrangeira referente a tblplayercharacter) | | Código jogador desafiante | Inteiro | Código do jogador desafiante | Sim |
| hostchallenger | | Hostname/IP desafiante | Texto | IP/hostname do jogador que criou a sala. | Sim |
| Idplayercharacterwinner | | Código jogador vencedor | Inteiro | Código do jogador vencedor | Sim |
| won\_at | | Data de vitória | Hora/data | Data de encerramento da partida. | Sim |
| created\_at | | Criado em | Hora/data | Data de criação da sala. | Sim |

## Script de criação e população das tabelas

/\*

SQLyog Enterprise - MySQL GUI v8.17

MySQL - 5.1.40-community : Database - stringsfighters

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

/\*!40101 SET NAMES utf8 \*/;

/\*!40101 SET SQL\_MODE=''\*/;

/\*!40014 SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0 \*/;

/\*!40014 SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0 \*/;

/\*!40101 SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='NO\_AUTO\_VALUE\_ON\_ZERO' \*/;

/\*!40111 SET @OLD\_SQL\_NOTES=@@SQL\_NOTES, SQL\_NOTES=0 \*/;

/\*Table structure for table `tblcharactertype` \*/

DROP TABLE IF EXISTS `tblcharactertype`;

CREATE TABLE `tblcharactertype` (

`idcharactertype` int(11) NOT NULL AUTO\_INCREMENT COMMENT 'Primary key.',

`idclan` int(11) DEFAULT NULL COMMENT 'Foreign key for tblclan.',

`title` varchar(50) DEFAULT NULL COMMENT 'Title for a character type.',

PRIMARY KEY (`idcharactertype`),

KEY `FK\_tblcharactertype\_idclan` (`idclan`),

CONSTRAINT `FK\_tblcharactertype\_idclan` FOREIGN KEY (`idclan`) REFERENCES `tblclan` (`idclan`) ON DELETE SET NULL ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8;

/\*Data for the table `tblcharactertype` \*/

insert into `tblcharactertype`(`idcharactertype`,`idclan`,`title`) values (1,1,'Dobot'),(2,2,'Ankh');

/\*Table structure for table `tblclan` \*/

DROP TABLE IF EXISTS `tblclan`;

CREATE TABLE `tblclan` (

`idclan` int(11) NOT NULL AUTO\_INCREMENT COMMENT 'Primary key.',

`title` varchar(50) DEFAULT NULL COMMENT 'Title of clan',

PRIMARY KEY (`idclan`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8;

/\*Data for the table `tblclan` \*/

insert into `tblclan`(`idclan`,`title`) values (1,'Letras'),(2,'Símbolos');

/\*Table structure for table `tblgameroom` \*/

DROP TABLE IF EXISTS `tblgameroom`;

CREATE TABLE `tblgameroom` (

`idgameroom` int(11) NOT NULL AUTO\_INCREMENT COMMENT 'Primary key.',

`idplayercharacterowner` int(11) DEFAULT NULL COMMENT 'Player id gameroom´s owner.',

`hostowner` varchar(15) DEFAULT NULL COMMENT 'Owner´s host IP',

`idplayercharacterchallenger` int(11) DEFAULT NULL COMMENT 'Player id of a challenger',

`hostchallenger` varchar(15) DEFAULT NULL COMMENT 'Challenger´s IP',

`idplayercharacterwinner` int(11) DEFAULT NULL COMMENT 'Player ID of a winner.',

`won\_at` datetime DEFAULT NULL COMMENT 'Winning date',

`created\_at` datetime DEFAULT NULL COMMENT 'Create date',

PRIMARY KEY (`idgameroom`),

KEY `FK\_tblgameroom\_idplayercharacterowner` (`idplayercharacterowner`),

KEY `FK\_tblgameroom\_idplayercharacterchallenger` (`idplayercharacterchallenger`),

CONSTRAINT `FK\_tblgameroom\_idplayercharacterowner` FOREIGN KEY (`idplayercharacterowner`) REFERENCES `tblplayercharacter` (`idplayercharacter`) ON DELETE SET NULL ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8;

/\*Table structure for table `tblplayer` \*/

DROP TABLE IF EXISTS `tblplayer`;

CREATE TABLE `tblplayer` (

`idplayer` int(11) NOT NULL AUTO\_INCREMENT COMMENT 'Primary key.',

`score` int(11) DEFAULT '0' COMMENT 'Player´s score',

`username` varchar(16) DEFAULT NULL COMMENT 'Player´s Username',

`password` varchar(16) DEFAULT NULL COMMENT 'Player´s password',

`active` smallint(1) DEFAULT '0' COMMENT 'Player´s active',

PRIMARY KEY (`idplayer`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8;

/\*Table structure for table `tblplayercharacter` \*/

DROP TABLE IF EXISTS `tblplayercharacter`;

CREATE TABLE `tblplayercharacter` (

`idplayercharacter` int(11) NOT NULL AUTO\_INCREMENT COMMENT 'Primary key.',

`idplayer` int(11) DEFAULT NULL COMMENT 'Foreign key for tblplayer.',

`idcharactertype` int(11) DEFAULT NULL COMMENT 'Foreign key for tblcharactertype.',

`level` int(11) DEFAULT '0' COMMENT 'Character´s Level',

`xp` int(11) DEFAULT '0' COMMENT 'Character´s XP',

`created\_at` datetime DEFAULT NULL COMMENT 'Create date',

PRIMARY KEY (`idplayercharacter`),

KEY `FK\_tblplayercharacter\_idcharactertype` (`idcharactertype`),

KEY `FK\_tblplayercharacter\_idplayer` (`idplayer`),

CONSTRAINT `FK\_tblplayercharacter\_idcharactertype` FOREIGN KEY (`idcharactertype`) REFERENCES `tblcharactertype` (`idcharactertype`) ON DELETE SET NULL ON UPDATE CASCADE,

CONSTRAINT `FK\_tblplayercharacter\_idplayer` FOREIGN KEY (`idplayer`) REFERENCES `tblplayer` (`idplayer`) ON DELETE SET NULL ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8;

/\*!40101 SET SQL\_MODE=@OLD\_SQL\_MODE \*/;

/\*!40014 SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS \*/;

/\*!40014 SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS \*/;

/\*!40111 SET SQL\_NOTES=@OLD\_SQL\_NOTES \*/;