

**INSTITUTO TÉCNICO DE BARUERI
BRASÍLIO FLORES DE AZEVEDO
CURSO TÉCNICO EM INFORMÁTICA**

Arthur Porto Silva 25318

Diego Gomes Nogueira da Silva 25346

Gabriel Leandro Santos Sousa 25283

Lucas Camargo Cancio 25264

Marcos Vinicius dos Santos Pereira 25242

CASA DE SHOWS DE STAND-UP

**BARUERI
DEZEMBRO / 2018**

Arthur Porto Silva – 25318
Diego Gomes Nogueira da Silva – 25346
Gabriel Leandro Santos Sousa – 25283
Lucas Camargo Cancio – 25264
Marcos Vinicius dos Santos Pereira – 25242

Turma: INF3AM

2018

CASA DE SHOWS DE STAND-UP

Trabalho de conclusão de curso apresentado
ao Instituto Técnico de Barueri “Brasílio
Flores de Azevedo” para a conclusão do
curso Técnico em Informática.
Orientadores: Professores Técnicos 3^a série

Barueri
2018

Arthur Porto Silva - 25318
Diego Gomes Nogueira da Silva - 25346
Gabriel Leandro Santos Sousa - 25283
Lucas Camargo Cancio - 25264
Marcos Vinicius dos Santos Pereira - 25242

CASA DE SHOWS DE STAND-UP

Trabalho de Conclusão de Curso apresentado
ao Instituto Técnico de Barueri Brasílio Flores
de Azevedo para a conclusão do Curso
Técnico em Informática em _____

Para a conclusão do Trabalho de Conclusão de Curso, os alunos participaram de um Projeto Interdisciplinar onde desenvolveram suas habilidades, competências e potencialidades individuais, propiciando o surgimento de profissionais empreendedores, capazes de implantar novas técnicas de gestão, métodos e processos inovadores.

O aluno acima qualificado realizou no período letivo o projeto descrito neste documento, cumprindo assim todas as etapas para sua aprovação.

Aprovado em ____/____/_____

COORDENAÇÃO CURSO DE INFORMÁTICA

Prof. Wagner Gusmão.
ITB Brasílio Flores de Azevedo

**Dedicamos este título ao integrante Lucas
Camargo Cancio responsável pela
programação por ter realizado a maior
parte do trabalho com a devida
responsabilidade e cuidado, priorizando as
necessidades impostas pelo grupo.**

AGRADECIMENTOS

Primeiramente a Deus, pois sem ele não teríamos chegado até aqui. Aos nossos familiares, por sempre terem nos aconselhado e ajudado quando precisamos. À todos os professores técnicos, por terem sempre nos ajudado com as dúvidas e problemas, além de terem nos dado a devida atenção quando precisamos. Aos profissionais técnicos da instituição, por terem nos auxiliado com o fluxo de arquivos requisitados. Aos nossos colegas de classe, por terem dado dicas, auxílio e informações que foram essenciais para a realização do projeto. Ao ITB, pela oportunidade de aprendizado. Nossos sinceros agradecimentos.

“A inovação no software, como em quase todos os outros tipos de inovação, requer a capacidade de colaborar e partilhar ideias com outras pessoas, e sentar e falar com os clientes, obter os seus comentários e perceber suas necessidades.”

Bill Gates

RESUMO

O projeto envolve a criação de um sistema Desktop para gerenciamento de uma casa de shows, um Website e um aplicativo Mobile para a divulgação e interação da mesma para com seus clientes.

Após uma longa negociação com o estabelecimento, o grupo decidiu, em consenso, a criação do projeto “Comedy House” para nosso cliente, que é uma casa de shows de Stand-Up Comedy situada em São Paulo.

A casa sofria com a perda de informações no processo de gerenciamento, pois todos os dados eram armazenados em papéis. Por isso fomos contatados para auxiliar na modernização da casa, o que ajudou a Comedy House a conquistar seu espaço na competição com outros estabelecimentos de grande porte.

Palavras-chave: Desktop; Website; Mobile; Comedy House.

ABSTRACT

The project involves the creation of a Desktop system for the management of a humor show house, a Website and a Mobile app for their divulgence and interaction with their clients.

After a long negotiation with the establishment, the group decided, in consensus, to start the “Comedy House” project for our client, which is a stand-up comedy house located in São Paulo.

The establishment was suffering with the information loss, because all the data were saved in papers only. Then, we were called to help them on their modernization, which helped Comedy House to get their space on the competition against other high renowned establishments.

Word-Key: Desktop; Website; Mobile; Comedy House.

LISTA DE FIGURAS OU IMAGENS

DESKTOP

Imagen 01 – Login	45
Imagen 02 – Pergunta Secreta.....	46
Imagen 03 – Hub	46
Imagen 04 – Procura de Assento	47
Imagen 05 – Venda de Ingresso	48
Imagen 06 – Cadastro de Eventos	49
Imagen 07 – Cadastro de Eventos - Artistas	50
Imagen 08 – Cadastro de Artistas.....	51
Imagen 09 – Cadastro de Empresário	52
Imagen 10 – Gerenciamento de Empresários.....	53
Imagen 11 – Gerenciamento de Artistas	54
Imagen 12 – Gerenciamento de Eventos.....	55
Imagen 13 – Log	55
Imagen 14 – Gerenciamento de Usuários	56
Imagen 15 – Novidades	57
Imagen 16 – Formas de Pagamento.....	57
Imagen 17 – Funções	58
Imagen 18 – Pagamento de Ingressos	58
Imagen 19 – Cadastro de Contas	59
Imagen 20 – Gerenciamento de Contas	60
Imagen 21 – Relatório de Finanças.....	61
Imagen 22 – Cadastro de Funcionários.....	62
Imagen 23 – Gerenciamento de Funcionários	62
Imagen 24 – Loading	63
Imagen 25 – Descontos	63
Imagen 26 – Gestão de Preços	64
Imagen 27 – Consulta de Ingressos	65
Imagen 28 – Consulta de Ingressos – Eventos	65
Imagen 29 – Consulta de Ingressos	66
Imagen 30 – Consulta de Ingressos – Cliente.....	66
Imagen 31 – Consulta de QRCode	67
Imagen 32 – Reembolso de Ingressos.....	67
Imagen 33 – Gerenciamento de Ingressos.....	68
Imagen 34 – Cadastro de Clientes	68
Imagen 35 – Consulta de Clientes	69
Imagen 36 – Gerenciamento de Clientes.....	69

WEB

Imagen 37 – Cabeçalho.....	129
Imagen 38 – Cabeçalho Logado	129
Imagen 39 – Rodapé	130
Imagen 40 – Home	131
Imagen 41 – Cadastro	131
Imagen 42 – Recuperação de Senha	132
Imagen 43 – Login	132
Imagen 44 – Artistas.....	133
Imagen 45 – MostrarArtista1	133
Imagen 46 – MostrarArtista2	134
Imagen 47 – Eventos	134
Imagen 48 – Mostrar Evento.....	135
Imagen 49 – Comprar Ingresso	136
Imagen 50 – Pagamento1	137

Imagen 51 – Pagamento2.....	138
Imagen 52 – Ajuda1	139
Imagen 53 – Ajuda2	139
Imagen 54 – Feedback	140
Imagen 55 – Feedback Logado	140
Imagen 56 – Perfil1	141
Imagen 57 – Perfil2	141

MOBILE

Imagen 58 – Login	144
Imagen 59 – Criar Usuário 1	145
Imagen 60 – Criar Usuário 2.....	146
Imagen 61 – Criar Usuário 3.....	147
Imagen 62 – Pergunta Secreta.....	148
Imagen 63 – Histórico	149
Imagen 64 – Consultar Evento.....	150
Imagen 65 – Hub1	151
Imagen 66 – Hub2	152
Imagen 67 – Hub3	153
Imagen 68 – Perfil.....	154
Imagen 69 – QR Code	155

LISTA DE CLASSES – DESKTOP

Imagen 70 – Classe Artista	33
Imagen 71 – Classe Assento	34
Imagen 72 – Classe CEP.....	34
Imagen 73 – Classe Cliente.....	35
Imagen 74 – Classe Contas	36
Imagen 75 – Classe Descontos	36
Imagen 76 – Classe Empresario	37
Imagen 77 – Classe Login.....	38
Imagen 78 – Classe Funcionario	39
Imagen 79 – Classe Ingresso.....	40
Imagen 80 – Classe Evento.....	41
Imagen 81 – Classe FormaPagamento.....	41
Imagen 82 – Classe Funcao.....	42
Imagen 83 – Classe Log.....	42
Imagen 84 – Classe Novidades	42
Imagen 85 – Classe PerguntaSecreta	43
Imagen 86 – Classe Pessoa	43
Imagen 87 – Classe ValidarCNPJ	43
Imagen 88 – Classe ValidarCPF.....	43

LISTA DE CLASSES – WEB

Imagen 89 – Classe Artistas.....	121
Imagen 90 – Classe Assento	122
Imagen 91 – Classe Boleto.....	122
Imagen 92 – Classe CEP.....	122
Imagen 93 – Classe Cliente.....	123
Imagen 94 – Classe Contas	124
Imagen 95 – Classe Descontos	124
Imagen 96 – Classe Login.....	125

Imagen 97 – Classe Ingresso.....	126
Imagen 98 – Classe Evento.....	127
Imagen 99 – Classe Feedback	127
Imagen 100 – Classe Pessoa	128
Imagen 101 – Classe ValidadorCPF.....	128

LISTA DE TABELAS

Tabela 1 – Dicionário de dados	16
Tabela 2 – Funções dos participantes	70
Tabela 3 – Dicionário de dados	76
Tabela 4 – Dicionário de dados	143
Tabela 5 – Funções dos participantes novos	156

LISTA DE SIGLAS

ABNT – Associação Brasileira de Normas Técnicas

PMB – Prefeitura Municipal de Barueri

C# - C Sharp

CSS - Cascading Style Sheets

SQL - Structured Query Language

UML - Unified Modeling Language

ADO.Net - ActiveX Data Objects

CPF - Cadastro de Pessoa Física

CNPJ - Cadastro Nacional da Pessoa Jurídica

ITB - Instituto Técnico de Barueri

SUMÁRIO

1	INTRODUÇÃO	12
2	SISTEMA COM ACESSO AO BANCO DE DADOS (C#.NET + ADO.NET).....	13
2.1	TERMO DE ABERTURA DO PROJETO	13
2.2	DESCRIÇÃO DO AMBIENTE	13
2.2.1	<i>Descrição do Processo / Problema.....</i>	13
2.2.2	<i>Descrição do Ambiente / Infraestrutura.....</i>	14
2.2.3	<i>Escopo do Sistema.....</i>	15
2.2.4	<i>Funções do Sistema</i>	15
2.2.5	<i>Diagrama de Contexto (DFD Nível 0).....</i>	15
2.3	DADOS DO SISTEMA	16
2.3.1	<i>Dados Armazenados</i>	16
2.3.2	<i>Dicionário de Termo de Dados</i>	16
2.3.3	<i>Diagrama de Entidade e Relacionamento (DER).....</i>	30
2.4	UML	31
2.4.1	<i>Diagrama de Caso de Uso</i>	31
2.4.2	<i>Caso de Uso</i>	31
2.4.3	<i>Diagrama de Classes.....</i>	32
2.4.4	<i>Representação das Classes</i>	32
2.5	PLANOS DE MANUTENÇÃO.....	44
2.5.1	<i>Planos de Manutenção.....</i>	44
2.5.2	<i>Agendamentos</i>	44
2.6	TELAS DO SISTEMA.....	45
2.6.1	<i>Interface do Sistema.....</i>	45
3	SISTEMA WEB DINÂMICO (ASP.NET)	70
3.1	GERENCIAMENTO DE PROJETO	70
3.1.1	<i>Termo de Abertura do Projeto (TAP)</i>	70
3.1.2	<i>Estrutura Analítica do Projeto (EAP).....</i>	71
3.1.3	<i>Cronograma.....</i>	72
3.2	ESCOPO.....	72
3.2.1	<i>Objetivo do Documento</i>	72
3.3	DESCRIÇÃO DO AMBIENTE	72
3.3.1	<i>Descrição do Processo / Problema.....</i>	72
3.3.2	<i>Descrição do Ambiente / Infraestrutura.....</i>	73
3.3.3	<i>Metas e Objetivos do Sistema.....</i>	74
3.3.4	<i>Sistemas atualmente em uso.....</i>	74
3.3.5	<i>Escopo do Sistema.....</i>	74
3.3.6	<i>Funções do Sistema</i>	74
3.3.7	<i>Regras de Negócio.....</i>	74
3.3.8	<i>DFD – Diagrama de Fluxo de Dados.....</i>	75
3.3.9	<i>DFD Nível 0 - Diagrama de Contexto</i>	75
3.3.10	<i>DFD Nível 1</i>	75
3.4	DADOS DO SISTEMA	76
3.4.1	<i>Dados Armazenados</i>	76
3.4.2	<i>Dicionário de Termo de Dados</i>	76
3.4.3	<i>Diagrama de Entidade e Relacionamento (DER).....</i>	76
3.4.4	<i>Script de criação do Banco de Dados (SQL).....</i>	76
3.5	UML	
119		
3.5.1	<i>Diagrama de Caso de Uso</i>	119

3.5.2	<i>Caso de Uso</i>	119
3.5.3	<i>Diagrama de Classes</i>	120
3.5.4	<i>Representação das Classes</i>	120
3.6	INTERFACE.....	129
3.6.1	<i>Telas e Navegação</i>	129
4	SISTEMA PARA DISPOSITIVOS MÓVEIS (ANDROID)	142
4.1	ESCOPO GERAL DO SISTEMA.....	142
4.1.1	<i>Escopo do Sistema</i>	142
4.1.2	<i>Funções do Sistema</i>	142
4.2	DADOS DO SISTEMA.....	143
4.2.1	<i>Dados Armazenados</i>	143
4.3	TELAS.....	144
4.3.1	<i>Navegação das Telas</i>	144
5	CONCLUSÃO	157

1 INTRODUÇÃO

O projeto envolve a criação de um sistema Desktop para gerenciamento de uma casa de shows, um Website e um aplicativo Mobile para a divulgação e interação da mesma para com os seus clientes.

Após um longo processo de decisão, o grupo decidiu, em consenso, o tema Casa de Shows por este ser pouco comum e algo que todos da equipe possuíssem conhecimento e tivessem interesse.

Pesquisas indicam que há muita perda de informação com o uso de papéis no gerenciamento dos estabelecimentos, por isso fomos contatados para ajudar na criação e modernização da Comedy House, que busca seu espaço competindo com outros locais de grande porte.

O que foi desenvolvido em cada parte do projeto é:

Sistema Corporativo – É a ferramenta para o cliente gerenciar a empresa, ou seja, controlar a entrada e saída de informações sobre os cadastros (clientes, funcionários), pagamentos, os agendamentos da casa de show;

Web Site – Será o centro de publicidade do projeto, ou seja, nele será mostrado as informações do show (preço e horário) e será possível o cadastro do usuário para então, o mesmo poder efetuar a compra virtual dos ingressos, escolhendo assento, tipo de assento e classificação;

Aplicativo móvel (mobile) – No aplicativo móvel o usuário cadastrado, poderá consultar todos os futuros eventos da casa de show e os eventos do dia, poderá também consultar seu histórico de compras, e o usuário também fará o uso do QRCode.

As linguagens usadas são: C#, ASP.NET, JAVA, SQL, CSS, Javascript, AJAX.

Os programas usados são: Visual Studio 2017, Android Studio, Oracle e SQL Server 2017.

2 SISTEMA COM ACESSO AO BANCO DE DADOS (C#.Net + ADO.Net)

2.1 Termo de Abertura do Projeto

2.2 Descrição do Ambiente.

2.2.1 Descrição do Processo / Problema

A casa de shows possui vários processos que precisam ser realizados para que o sistema funcione corretamente. No sistema deverão ser cadastrados os artistas, eventos, funcionários, empresários, contas e ingressos. Todos eles são necessários porque um depende do outro.

Todas as funções e processos são controlados pela tela principal do sistema, onde tudo está disponível. Para o cadastro de artistas será necessário colocar suas informações e também a de um empresário. Caso ele seja um artista contratado, que no caso seria um artista da casa que recebe salário, ele não terá a opção de empresário. Após isso, ele será registrado no sistema.

Para o cadastro de eventos, deverá ser colocado um horário, data, escolher os artistas que irão participar e também as informações do evento.

Para os empresários deverá ser colocado as informações básicas como contatos e localização por parte das informações pessoais.

No cadastro de funcionários deverá ser colocado as informações básicas e escolher uma das funções para ele. Dependendo da função selecionada, o mesmo possuirá um login no sistema.

Para as contas, é escolhido se a conta será a pagar ou a receber, também é determinado uma descrição, departamento, data de lançamento, data de vencimento, situação, forma de pagamento, o valor pago e o valor nominal.

O processo mais importante do sistema é o da venda do ingresso. Primeiro o cliente deverá escolher o evento que deseja ir, e após isso ele deve escolher os assentos que ele vai querer comprar no esquema da sala. Finalmente, após o cliente decidir o evento e os assentos, ele é redirecionado para o pagamento, onde será mostrado o valor total, os descontos caso haja um, o troco, e a forma de pagamento. E então o ingresso é gerado.

Os principais erros e falhas no sistema se dão devido à pequenas falhas de programação, mas no geral são problemas que envolvem a análise do sistema e composição de cada processo do projeto.

2.2.2 Descrição do Ambiente / Infraestrutura

A Comedy House fica em um edifício de 1 andar (similar a um Teatro) e no total possui 1000 m², sendo constituído por:

- 6 Banheiros (feminino, masculino e deficiente) de 25 m² cada;
- 1 Lanchonete de 150 m²;
- 1 Palco de 36 m²;
- Um espaço de 594 m² para todos os assentos que são divididos em tipos (Duplo, Especial, Simples) categorias (Ouro, Prata, Bronze)
 - Ouro: Assentos mais próximos ao palco;
 - Prata: Assentos não tão próximos ao palco;
 - Bronze: Assentos distantes do palco;
- Recepção – Um espaço de 50 m² de recepção, lugar onde ficam os computadores, que são 3 no total;
- Camarim – Um espaço de 20 m² de camarim; O edifício tem a capacidade máxima de 170 espectadores e possui 20 funcionários, sendo 13 faxineiros(as), 3 recepcionistas, 4 seguranças.

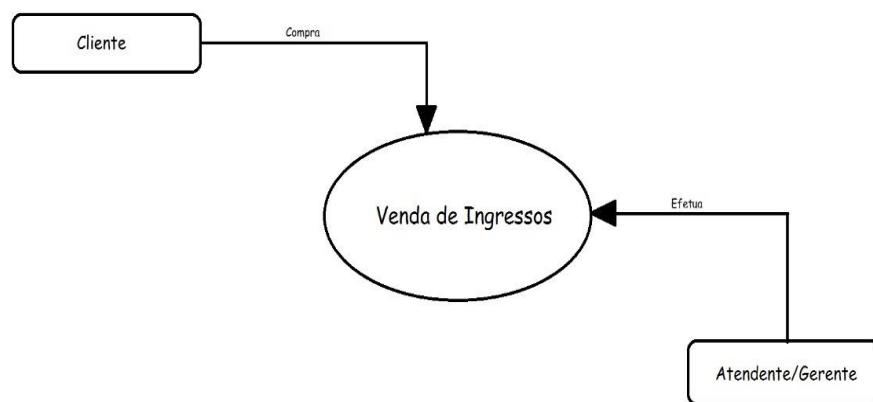
2.2.3 Escopo do Sistema

O escopo do sistema se baseia na venda do ingresso, pois ele é a característica principal do sistema, uma vez que a casa de show tem como principal fonte de renda os ingressos vendidos nos eventos, de modo que as pessoas vão comprá-los baseados nos setores que são lhes proporcionados. Por esta razão os preços são acessíveis, estando de acordo com a quantia no qual os clientes estão aptos a pagar, aumentando assim, a lucratividade do negócio e facilitando a venda dos mesmos. Os ingressos são simples e possuem as informações necessárias que o cliente precisa saber, como o evento, horário, local e os artistas. O software em que o sistema da venda do ingresso é usado é simples e objetivo, onde é seguido uma ordem lógica para que o ingresso seja gerado.

2.2.4 Funções do Sistema

A principal função é a venda de ingressos. Essa venda é feita seguindo uma lógica, que começa nas escolhas do evento e do assento, respectivamente, e em seguida se encerra no pagamento dos mesmos. O valor base de um evento pode ser determinado pelos próprios artistas ou pela gerência da casa.

2.2.5 Diagrama de Contexto (DFD Nível 0)



2.3 Dados do Sistema

2.3.1 Dados Armazenados

2.3.2 Dicionário de Termo de Dados

Tabela: TB_TIPO_PESSOA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_TIPO_PESSOA		Inteiro	10			Não nulo
TIPO_PESSOA	Função da pessoa	Texto	12	Funcionário, Cliente, Artista, Artista fixo		Não nulo

Tabela: TB_PESSOA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_PESSOA		Inteiro	10			Não nulo
NOME		Texto	50			Não nulo
DATA_NASC		Data	10		DD/MM/AAAA	
SEXO		Texto	25	Feminino/ Masculino		Não nulo
ATIVO		Numero	1	0, 1		Não nulo
ID_TIPO_PESSOA	Identificação do tipo da pessoa	Inteiro	10			Não nulo
DATA_CRIAÇÃO		Data	10		DD/MM/AAAA	

Tabela: TB_CEP

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
CEP		Texto	8		99999-999	Não nulo
UF	Unidade de federação	Texto	2			
DESCRIÇÃO		Texto	256			
CIDADE		Texto	27			
BAIRRO		Texto	128			

Tabela: TB_ENDERECHO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições

ID_ENDERECO		Inteiro	10			
CEP		Texto	8		99999-999	
NUMERO		Numero	6			
COMPLEMENTO	Casa, Apartamento, Condomínio	Texto	120			

Tabela: TB_AGENTE

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_AGENTE		Inteiro	10			Não nulo
ID_ENDERECO		Inteiro	10			
TELEFONE		Texto	14		(99)9999-9999	
TIPO_PESSOA	Função da pessoa	Texto	8			Não nulo
DOCUMENTO	CPF ou CNPJ	Texto	18			
EMAIL		Texto	120		AA@aaa.com	
NOME_PRINCIPAL	Nome real	Texto	120			
NOME_SECUNDARIO	Razão social	Texto	120			
ATIVO		Número	1	0, 1		
DATA_CRIAÇÃO		Data	10		DD/MM/AAA	

Tabela: TB_REDESOCIAL

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_REDESOCIAL		Inteiro	10			Não nulo
FACEBOOK		Texto	120			
TWITTER		Texto	120			
INSTAGRAM		Texto	120			

Tabela: TB_ARTISTA_GERAL

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_ARTISTA_GERAL		Inteiro	10			Não nulo

ID_REDESOCIAL		Inteiro	10			
ID_PESSOA		Inteiro	10			Não nulo
URL_IMG	Imagen	Texto	250			

Tabela: TB_ARTISTA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_AGENTE		Inteiro	10			Não nulo
ID_ARTISTA_GERAL		Inteiro	10			Não nulo

Tabela: TB_ARTISTA_FIXO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_ARTISTA_GERAL		Inteiro	10			Não nulo
ID_ENDERECHO		Inteiro	10			Não nulo
TELEFONE		Texto	14		(99)9999-9999	
CPF		Texto	12		999.999.999-99	
EMAIL		Texto	120		aaa@aa.com	

Tabela: TB_FORMA_PAGAMENTO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_FORMA_PAGAMENTO		Inteiro	10			Não nulo
FORMA_PAGAMENTO	Cartão de crédito, Cartão de débito, etc	Texto	30			Não nulo

Tabela: TB_DATA_CONTA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições

ID_DATA_CONTA		Inteiro	10			Não nulo
DATA_CONTA		Data	8		DD/MM/AA	
TIPO_DATA		Texto	11			Não nulo

Tabela: TB_CONTAS

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_CONTAS		Inteiro	10			Não nulo
ID_DATA_CONTA		Inteiro	10			Não nulo
TIPO_CONTA		Texto	15			
DATA_LANCAMENTO		Data	8		DD/MM/AA	
DATA_ENTREGUE		Data	8	Pagamento/ Recebimento	DD/MM/AA	
DESCRICAO		Texto	220			
SITUACAO		Texto	30			
DEPARTAMENTO		Texto	30			
VALOR_TOTAL		Número	24,2		R\$9999,99	
ATIVO		Número	1	0, 1		

Tabela: TB_DETALHE_CONTA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_DETALHE		Inteiro	10			
ID_CONTAS		Inteiro	10			
ID_FORMA_PAGAMENTO		Inteiro	10			
VALOR		Número	24,2		R\$9999,99	

Tabela: TB_CLIENTE

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições

ID_CLIENTE		Inteiro	10			Não nulo
CPF		Texto	12		999.999.999-99	
ID_PESSOA		Inteiro	10			Não nulo
EMAIL		Texto	120		aaa@aaa.com	

Tabela: TB_TIPO_ASSENTO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_TIPO_ASSENTO		Inteiro	10			Não nulo
TIPO_ASSENTO		Texto	8	Simples/Especial/ Conjugado		

Tabela: TB_SETOR

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_SETOR		Inteiro	10			Não nulo
SETOR		Texto	6	Ouro/Prata/Bronze		

Tabela: TB_ASSENTO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_ASSENTO		Inteiro	10			Não nulo
ASSENTO_NUMERO	Número do assento	Texto	20			Não nulo
ASSENTO_FILEIRA	Fileira do assento	Texto	1			Não nulo
ID_TIPO_ASSENTO		Inteiro	10			Não nulo
ID_SETOR		Inteiro	10	Ouro/Prata/Bronze		Não nulo

Tabela: TB_ASSENTO_CLI

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_ASSENTO		Inteiro	10			Não nulo

ID_CLIENTE		Inteiro	10			Não nulo
ID_EVENTO		Inteiro	10			Não nulo
ATIVO		Numero	1	0, 1		Não nulo

Tabela: TB_EVENTO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_EVENTO		Inteiro	10			Não nulo
TITULO		Texto	20			
DATA_EVENTO		Data	8		DD/MM/AAA A	
DESCRICAO		Texto	141		99:99	
HORARIO_INICIO		Texto	4		99:99	
HORARIO_FINAL		Texto	4			
DURACAO		Texto	5			
N_ARTISTA		Numero	3			
TIPO_PAG		Texto	20			
ATIVO		Numero	1	0, 1		Não nulo
VALOR_EVENTO		Numero	10,2		R\$999,99	
URL_IMG		Texto	250			

Tabela: TB_EVENTO_INGRESSO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições

ID_EI	ID_EVENTO_INGRESSO	Inteiro	10			Não nulo
ID_EVENTO		Inteiro	10			
ID_PAGAMENTO		Inteiro	10			
PRECO_TOTAL		Número	50		R\$999,99	
DESCONTO		Número	50		R\$999,99	
PRECO_TOTAL_PA GO		Número	50		R\$999,99	
TROCO		Número	50		R\$999,99	

Tabela: TB_VENDA_INGRESSO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_VENDA		Inteiro	10			Não nulo
ID_PESSOA		Inteiro	10			
DATAVENDA		Data	8		DD/MM/AAA A	
SITUACAO		Texto	15			
ATIVO		Número	1	0, 1		

Tabela: TB_DETALHE_CONTA_VENDA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_DETALHE_VENDA		Inteiro	10			Não nulo
ID_CONTAS		Inteiro	10			Não nulo
ID_VENDA		Inteiro	10			Não nulo

Tabela: TB_ITEM_VENDA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_ITEM_VENDA		Inteiro	10			Não nulo

ID_VENDA		Inteiro	10			Não nulo
ID_EI	ID EVENTO INGRESSO	Inteiro	10			Não nulo
ID_ASSENTO		Inteiro	10			Não nulo

Tabela: TB_PAGAMENTO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_PAGAMENTO_GERAL		Inteiro	10			Não nulo
ID_PAGAMENTO		Inteiro	10			Não nulo
ID_FORMA_PAGAMENTO		Inteiro	10			
QUANTIA		Número	38,5			

Tabela: TB_EVENTO_ARTISTA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_EVENTO		Inteiro	10			Não nulo
ID_PESSOA		Inteiro	10			Não nulo

Tabela: TB_PERGUNTASECRETA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_PERGUNTASECRETA		Inteiro	10			Não nulo
DESCRICAO		Texto	128			
RESPOSTA		Texto	120			

Tabela: TB_NIVELACESSO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições

ID_NIVELACESSO		Inteiro	10			Não nulo
NIVELACESSO		Texto	20	Gerente/Atendente		Não nulo

Tabela: TB_LOGIN

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_LOGIN		Inteiro	10			Não nulo
ID_NIVELACESSO		Inteiro	10			Não nulo
ID_PERGUNTASECERETA		Inteiro	10			Não nulo
USUARIO		Texto	50			Não nulo
SENHA		Texto	50		*****	Não nulo
DATAULTIMOACESSO		Data	8		DD/MM/AA AA	

Tabela: TB_FUNCAO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_FUNCAO		Inteiro	10			Não nulo
FUNCAO		Texto	40			Não nulo

Tabela: TB_FUNCIONARIO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições

ID_FUNC	ID FUNCIONARIO	Inteiro	10			Não nulo
ID_FUNCAO		Inteiro	10			Não nulo
ID_ENDERECO		Inteiro	10			
ID_PESSOA		Inteiro	10			Não nulo
CPF		Texto	12		999.999.999 -99	
EMAIL		Texto	120		aaa@aa.com	
TELEFONE		Texto	14		(99)9999-9999	

Tabela: TB_CANCELAVENDA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_CANCELAVENDA		Inteiro	10			Não nulo
ID_VENDA		Inteiro	10			Não nulo
MOTIVO		Texto	20			
DATA_CANCELAMENTO		Data	8		DD/MM/AA AA	
ID_FUNC	ID FUNCIONARIO	Inteiro	10			Não nulo

Tabela: TB_PRECOS

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_SETOR		Inteiro	10			Não nulo
ID_TIPO_ASSENTO		Inteiro	10			Não nulo
PERC	?	Numero	3		?	

Tabela: TB_BANNER

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_BANNER		Inteiro	10			Não nulo

URL_IMG		Texto	250			
FILE NAME		Texto	250		?	
INDEX_BANNER		Numero	12		?	
ATIVO		Numero	1	0, 1		

Tabela: TB_DESCONTOS

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_DESCONTOS		Inteiro	10			Não nulo
DESCONTO		Número	5,1			
COD_PROMOCIONAL		Texto	20			
ATIVO		Número	1			
N_USOS		Número	5			

Tabela: TB_DESCONTO_CLIENTE

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_DESCONTO		Inteiro	10			Não nulo
ID_CLIENTE		Inteiro	10			Não nulo

Tabela: TB_TIPO_LOG

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_TIPO_LOG		Inteiro	10			Não nulo
TIPO_LOG		Texto	15			

Tabela: TB_LOG

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições

ID_LOG		Inteiro	10			Não nulo
DATA_LOG		Data	8		DD/MM/AAA A	
ID_TIPO_LOG		Inteiro	10			Não nulo
ID_FUNC	ID FUNCIONARIO	Inteiro	10			Não nulo

Tabela: TB_TABELAS_LOG

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_TABELA_LOG		Inteiro	10			Não nulo
TABELA		Texto	15			
TABELA_APELIDO		Texto	15			

Tabela: TB_LOG_INSERCAO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_LOG_INSER		Inteiro	10			Não nulo
ID_LOG		Inteiro	10			Não nulo
ID_TABELA_LOG		Inteiro	10			Não nulo
ID_INSERIDO		Inteiro	10			Não nulo

Tabela: TB_LOG_EXCLUSAO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_LOG_EXCL		Inteiro	10			Não nulo
ID_LOG		Inteiro	10			Não nulo
ID_TABELA_LOG		Inteiro	10			Não nulo
ID_EXCLUIDO		Inteiro	10			Não nulo
DESCRICAO		Texto	50			

Tabela: TB_LOG_ALTERACAO

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_LOG_ALTER		Inteiro	10			Não nulo
ID_LOG		Inteiro	10			Não nulo
ID_TABELA_LOG		Inteiro	10			Não nulo
ID_ALTERADO		Inteiro	10			Não nulo

Tabela: TB_BANDEIRA

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_BANDEIRA		Inteiro	10			Não nulo

DESC_BANDEIRA	Descrição da bandeira	Texto	20			Não nulo
---------------	-----------------------	-------	----	--	--	----------

Tabela: TB_DETALHE_PGTO_CRED

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_PGTO_CRED	ID PAGAMENTO CRÉDITO	Inteiro	10			Não nulo
ID_PAGAMENTO		Inteiro	10			Não nulo
ID_PAGAMENTO_GERAL		Inteiro	10			Não nulo
ID_BANDEIRA		Inteiro	10			Não nulo
QTDE_PARCELAS	Quantidade de parcelas	Número	2			Não nulo

Tabela: TB_DETALHE_PGTO_DEB

Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_PGTO_DEB	ID PAGAMENTO DÉBITO	Inteiro	10			Não nulo
ID_PAGAMENTO		Inteiro	10			Não nulo
ID_PAGAMENTO_GERAL		Inteiro	10			Não nulo
ID_BANDEIRA		Inteiro	10			Não nulo

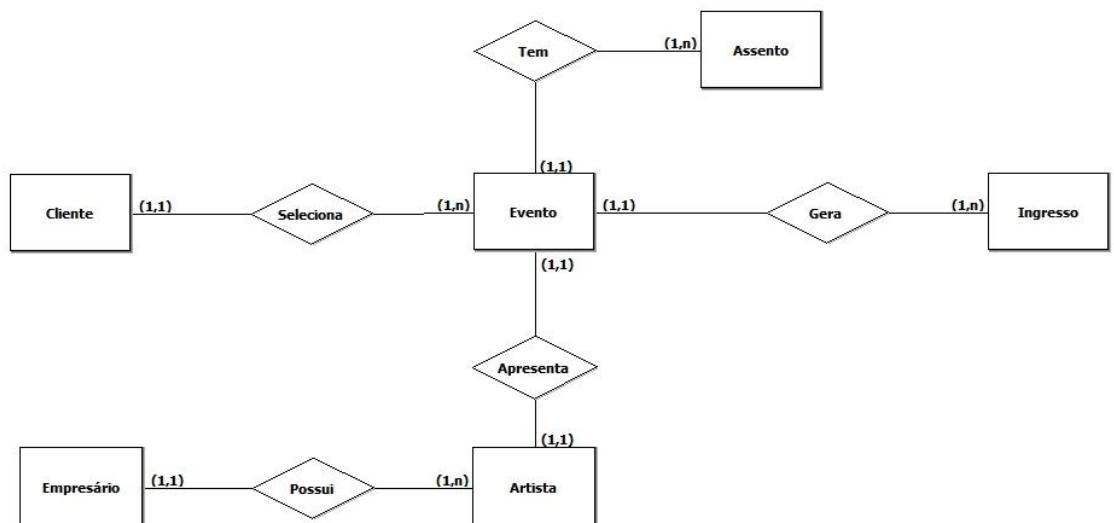
Tabela: TB_FEEDBACK

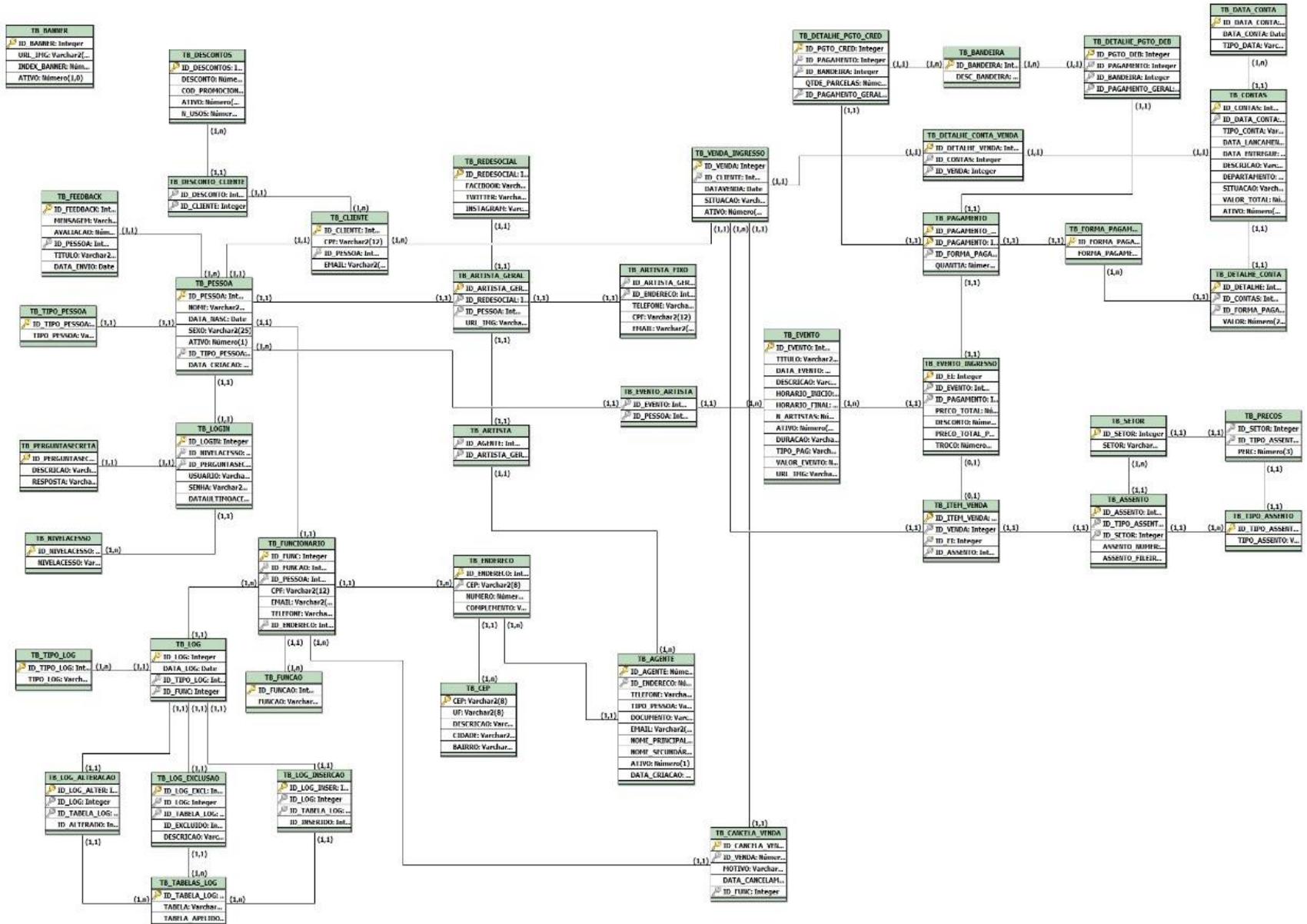
Nome	Descrição	Tipo	Tamanho	Domínio	Formato	Restrições
ID_FEEDBACK		Inteiro	10			Não nulo

MENSAGEM		Texto	120		
AVALIACAO		Numero	1		
ID_CLIENTE		Inteiro	10		Não nulo
TITULO		Texto	60		

Tabela 1 - Dicionário de Dados

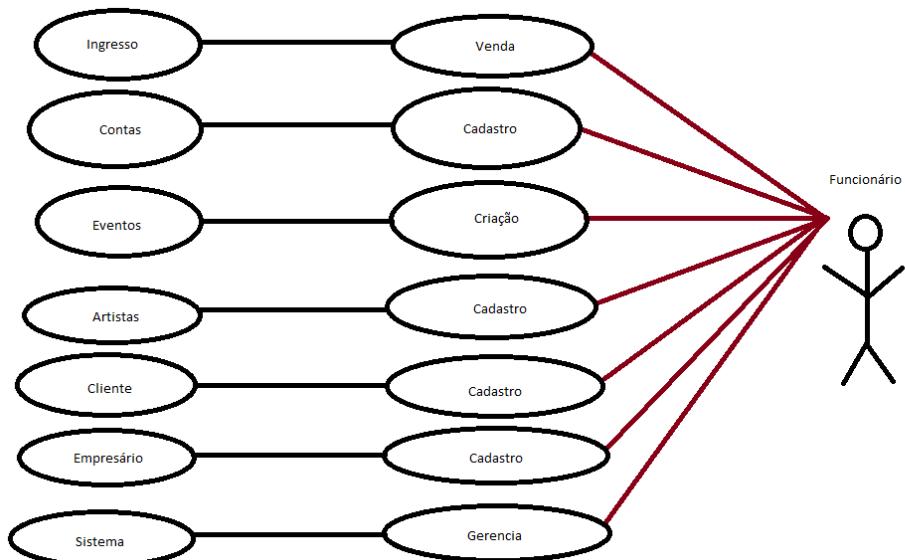
2.3.3 Diagrama de Entidade e Relacionamento (DER)





2.4 UML

2.4.1 Diagrama de Caso de Uso



2.4.2 Caso de Uso

O funcionário da casa poderá manipular várias das funções do sistema. Irá efetuar a venda dos ingressos para os clientes, cadastrará contas, artistas, clientes, empresários, gerenciará o sistema geral da casa e criará os eventos.

2.4.3 Diagrama de Classes



2.4.4 Representação das Classes

Artistas

Artista
ID_ARTISTA_GERAL: Int ID_TIPO_PESSOA: Int ID_PESSOA: Int ID_AGENTE: Int ID_ENDERECO: Int ID_REDESOCIAL: Int DATALISTARFINAL: date DATALISTARINICIAL: date SEXO: String URL_IMG: String TELEFONE: String NOME: String FACEBOOK: String TWITTER: String INSTAGRAM: String EMAIL: String SOCIAL: String DATADECRIACAO: date DATANASC: date ATIVO: Int CPF: String
ConsultarValorMaximo Verificar Ativar ConsultarPessoa ConsultarNormal Consultarfixo ListarGeral VerificarEvento ConsultarCPF Artista_crud

Imagen 70 – Classe Artista

Assento

Assento
ID_ASSENTO: Int
ID_TIPOASSENTO: Int
ID_SETOR: Int
VALOR: Double
ASSENTO_NUMER: Int
ASSENTO_FILEIRA: String
AlterarPrecos
ConsultarPercentual
ConsultarValor2
ConsultarAssento
ConsultarValorMaximo

Imagen 71 – Classe Assento

CEP

CEP
ID_ENDERECHO: Int
CEP: String
NUMERO: double
COMPLEMENTO: String
Consultar
Endereco_crud

Imagen 72 – Classe CEP

Cliente

Cliente
ID_CLIENTE: Int ID_ASSENTO: Int ID_EVENTO: Int ID_TIPO_PESSOA: Int ID_PESSOA:Int NOME: String DATADESCRICAOP: date DATALISTARINICIO: date DATALISTARFINAL: date DATANASC: date ATIVO: Int CPF: String EMAIL: String ConsultarPessoa ConsultarValorMaximo ConsultarPorCliente Consultar Ativar ListarGeral ListarHistorico ConsultarCPF Cliente_crud

Imagen 73 – Classe Cliente

Contas

Contas
ID_CONTAS: Int ID_VENDA: Int ID_DATACONTA: Int DATA: date DATALISTARFINAL: date DATALISTARINICIO: date DATA_ENTREGUE: date DATALANCAMENTO: date TIPO_CONTA: String TIPO_DATA: String DESCRICAO: String DEPARTAMENTO: String FORMA_PAGAMENTO: String SITUACAO: String VALOR_TOTAL: double ATIVO: Int ID_FORMA_PAGAMENTO: Int ConsultarValorMaximo ExcluirFormaPag IncluirVenda IncluirFormaPag Ativar ConsultarFormas ListarGeral ListarGrafico Consultar ConsltarVenda Contas_crud

Imagen 74 – Classe Contas

Descontos

Descontos
ID_DESCONTOS: Int ID_CLIENTE: Int DESCONTO: double CODPROMOCIONAL: String N_USOS: double ATIVO: Int Consultar ConsultarPorId ConsultarValorMaximo Listar Descontos_crud

Imagen 75 – Classe Descontos

Empresario

Empresario
ID_AGENTE: Int
ID_PESSOA: Int
ID_ENDERECHO: Int
DOCUMENTO: String
TELEFONE: String
EMAIL: String
NOMEPRINCIPAL: String
NOMESECUNDARIO: String
TIPOPESSOA: String
DATADECRIACAO: date
DATALISTARINICIO: date
DATALISTARFINAL: date
ATIVO: Int
Ativar
VerificarArtista
ConsutarGeral
ConsultarValorMaximo
ConsultarDocumento
ListarGeral
Agente_crud

Imagen 76 – Classe Empresario

Login

Login
ID_LOGIN: Int ID_PESSOA: Int ID_FUNC: Int ID_CLI: Int USUARIO: String NOME: String NIVEL_ACESSO: String ID_NIVELACESSO: Int SENHA: String DATAULTIMOACESSO: date PERGUNTASECRETAID: Int DESCRICAO: String ATIVO: Int DATALISTARINICIO: date DATALISTARFINAL: date
RecuperarDados TrocSenha ListarGeral AlterarSenha ConsultarCli Logar Login_crud SpIniciar ConsultarUsuario

Imagen 77 – Classe Login

Funcionario

Funcionario
ID_FUNCIONARIO: Int SEXO: String DATALISTARFINAL: date DATALISTARINICIO: date ID_FUNCAO: Int ID_ENDERECHO: Int ID_PESSOA: Int NOME: String ATIVO: Int DATA_NASC: date TELEFONE: String EMAIL: String CPF: String ConsultarPessoa ConsultarNomePessoa ConsultarValorMaximo Ativar Consultar ConsultarCPF Consultar2 ListarGeral Funcionario_crud

Imagen 78 – Classe Funcionario

Ingresso

Ingresso

ID_FUNC: Int
 ID_CLIENTE: Int
 MOTIVO: String
 ID_EVENTO: Int
 ID_EI: Int
 DATA LISTAR FINAL: date
 DATA LISTAR INICIO: date
 ID_VENDA: Int
 ID_PAGAMENTO: Int
 PRECO_TOTAL: Double
 SITUACAO: String
 PRECO_TOTAL_PAGO: Double
 DESCONTO: Double
 TROCO: Double
 ATIVO: Int

IncluirEventoIngresso
 IncluirPagamento
 IncluirVendaIngresso
 IncluirAssentoCliente
 IncluirItemVenda
 ValidarIngresso
 Cancelar
 ConsultarPorEvento
 ConsultarReembolso
 ConsultarPorCliente
 ConsultarValidade
 ListarIngresso
 ListarIngressoAssento
 ListarFormasPag
 ListarGrafico

Imagen 79 – Classe Ingresso

Evento

Evento
ID_EVENTO: Int ID_ARTISTA_GERAL: Int DATAEVENTO: date HORARIOINICIO: date HORARIOFINAL: date TITULO: String DATALISTARFINAL: date DATALISTARINICIO: date DURACAO: date DESCRICAO: String TIPO_PAG: String ATIVO: Int N_ARTISTAS: Double VALOR_EVENTO: double ConsultarValorMaximo IncluirEventoArtista AdicionarEventoArtista Ativar ExcluirArtista Consultar Verificar VerificarIngresso ListarID_e_TITULO ListarDIA ListarGeral ListarArtistasDoEvento Evento_crud

Imagen 80 – Classe Evento

FormaPagamento

FormaPagamento
ID_FORMA_PAGAMENTO: Int FORMA_PAGAMENTO: String ConsultarPorId ConsultarValorMaximo ListarCombobox Listar FormaDePag_crud

Imagen 81 – Classe FormaPagamento

Funcao

Funcao
ID_FUNCAO: Int
FUNCAO: String
ConsultarPorId
ConsultarValorMaximo
ListarComboBox
Listar
VerificarFuncionario
Funcao_crud

Imagen 82 – Classe Funcao

Log

Log
ID_LOG: Int
TIPO_LOG: String
DESCRICAOP: String
DATA_LOG: date
ID_FUNC: Int
ID_MODIFICADO: Int
TABELA_LOG: String
DATALISTARFINAL: date
DATALISTARINICIO: date
Listar
ConsultarExcluido
Log_crud

Imagen 83 – Classe Log

Novidades

Novidades
ID_BANNER: Int
URL_IMG: String
ATIVO: Int
INDEX_BANNER: String
MudarDeLugar
Consultar
ConsultarAtivo
ConsultarTamanho
Novidades_crud

Imagen 84 – Classe Novidades

PerguntaSecreta

	PerguntaSecreta
PERGNTASECRETAID: Int	
DESCRICAO: String	
RESPOSTA: String	
ATIVO: Int	
Consultar	

Imagen 85 – Classe PerguntaSecreta

Pessoa

	Pessoa
ID_PESSOA: Int	
NOME: String	
ID_TIPO_PESSOA: Int	
DATA_NASC: date	
DATA_CRIACAO: date	
SEXO: String	
ATIVO: Int	
Pessoa_crud	

Imagen 86 – Classe Pessoa

ValidadorCNPJ

	ValidadorCNPJ
	-
	ValidarCNPJ

Imagen 87 – Classe ValidadorCNPJ

ValidadorCPF

	ValidadorCPF
	-
	ValidarCPF

Imagen 88 – Classe ValidadorCPF

2.5 Planos de Manutenção

As manutenções do sistema se darão em caráter adaptativo, visando melhorar a solução fornecida, adequando os softwares em suas plataformas (Web, Desktop, Mobile) as condições as quais eles são utilizados, tornando-os cada vez mais adaptados e mais eficientes ao negócio.

Algumas rotinas como: Backups do Banco de Dados, mudança de parâmetros, adição de campos e tabelas, se dará periodicamente.

2.5.2 Agendamentos

A manutenção do sistema será feita mensalmente e caso após a implementação do Software surgirem problemas em uma das plataformas oferecidas, as manutenções de software em caráter corretivo serão utilizadas com um prazo de 24 horas.

2.6 Telas do Sistema

2.6.1 Interface do Sistema



Imagen 01 – Login

Login: Ponto de acesso para o sistema corporativo, onde são digitados o usuário e a senha. Também possui algumas funções para caso o usuário esqueça a senha, onde é possível alterá-la, como também é possível ser direcionado para o website da casa.



Imagen 02 – Esqueceu a senha

Esqueceu a Senha: Aqui o usuário deverá colocar a resposta da pergunta secreta, em seguida, será liberado o acesso para que se mude a senha.

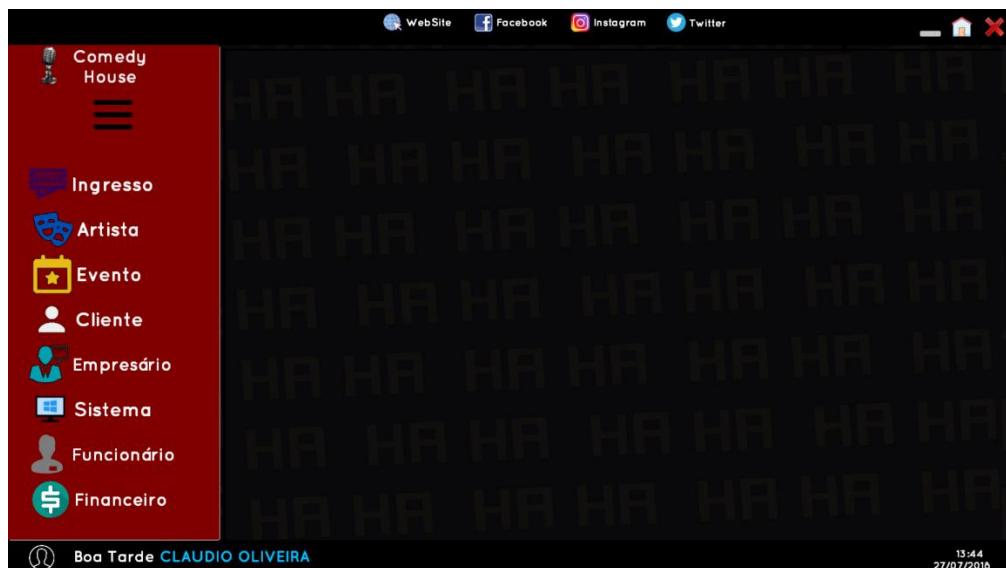


Imagen 03 – Hub

Hub: É a tela principal do sistema corporativo. Aqui é possível ter acesso à todas as funções do sistema, verificar o nível de acesso (representado pela cor do nome no canto inferior esquerdo) e ter acesso às redes sociais e o website da casa.

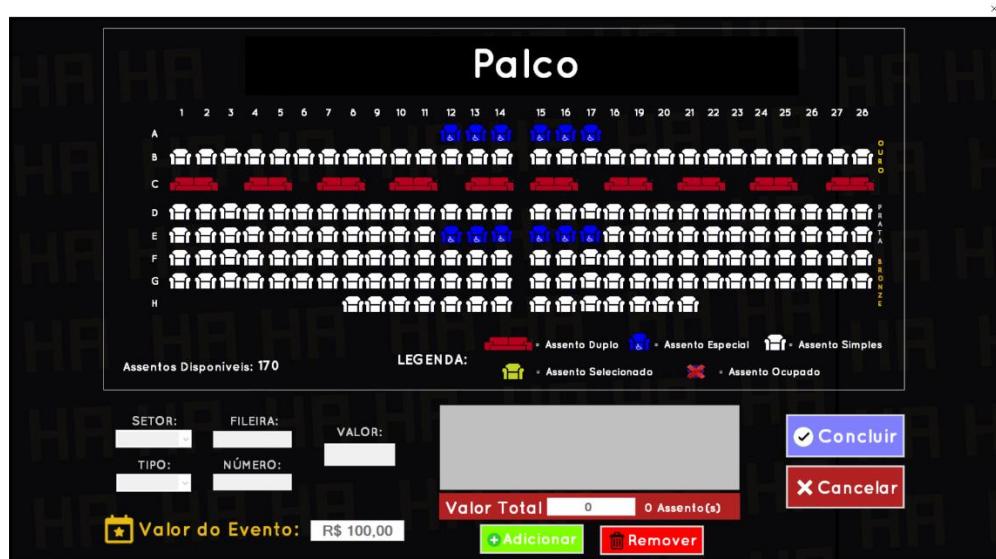


Imagen 04 – Procura de assento

Procura de Assento: Aqui é possível verificar o esquema de assentos da sala, verificar os assentos disponíveis, ver a legenda, escolher a categoria que deseja, a fileira, o tipo e o número do assento, podendo também se comprar até cinco assentos. Após isso, é gerado um valor que é adicionado ao grid e se acumula dependendo da quantidade de assentos escolhidos.

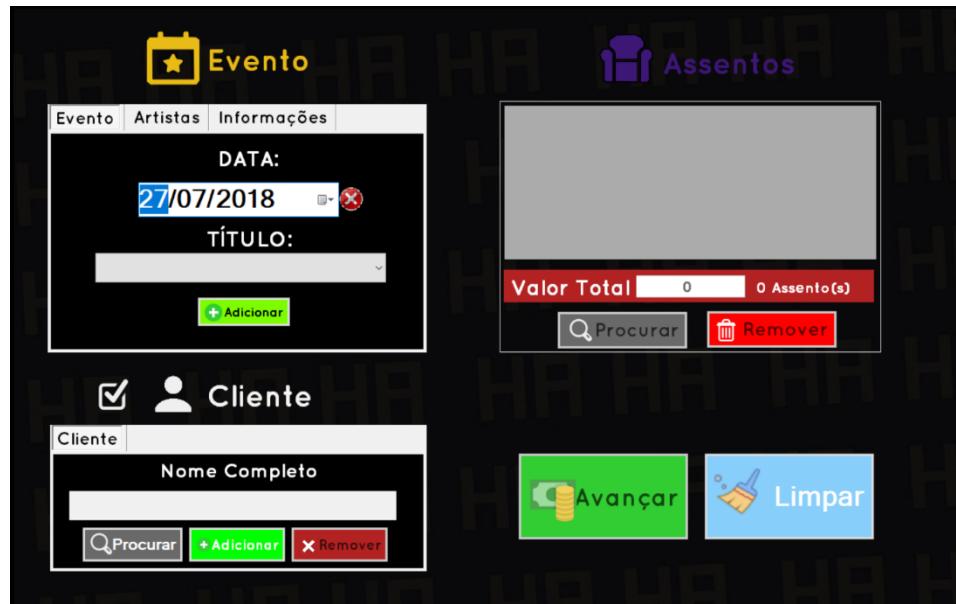


Imagen 05 – Venda de ingresso

Venda do Ingresso: Aqui é onde será escolhido um evento que vai estar no ingresso ao clicar em “Adicionar”. Além disso ele mostra os artistas e as informações do evento. Após escolhido o evento, você será redirecionado para a tela de procura de assentos, onde os mesmos serão escolhidos. Caso a pessoa seja ou queira se tornar um cliente da casa, ela tem essa opção para escolher habilitando a opção “Cliente”.

Cadastro de Eventos
Todos os campos com * são obrigatórios

Ativo

Informações | **Artistas**

Data do Evento: 27/07/2018

Horário * Duração: De 18:00 Até 00:30 Até 18:30

Eventos do Dia:

Cód	Título	Data Do Evento	Descrição	Horário Inicial	Horário Final

Valor do Evento *

Consultar **Limpar** **Salvar** **Cancelar**

Imagen 06 – Cadastro de eventos

Cadastro de Eventos: É a tela para cadastrar um evento no sistema, no qual são escolhidos seu horário, informações e artistas designados para o evento. Também é escolhido uma data e uma duração. O horário mínimo e máximo para cadastro são 18:00 e 00:00, respectivamente.



Imagen 07 – Cadastro de eventos – Artistas

Cadastro de Eventos – Artistas: Aqui é onde são escolhidos quais artistas vão participar de um evento. Um artista fixo não pode ser registrado em um evento junto com um artista chamado pela casa, por isso tem a escolha de um tipo de pagamento; Bilheteria para um artista chamado e um cachê fixo para um artista fixo.

Imagen 08 – Cadastro de artistas

Cadastro de Artista: Aqui é possível decidir o se o artista vai ser fixo (trabalha pela casa e recebe salário mensal) ou um artista de fora, onde digita-se o nome artístico do artista, o sexo, a data de nascimento (se quiser) e escolhe uma foto do artista. Na seção do empresário é feita uma busca ou um cadastro de um empresário. Também são colocadas as informações de contato e suas redes sociais. O código é gerado automaticamente e o artista é ativado logo após o término do cadastro.

Imagen 09 – Cadastro de empresário

Cadastro de Empresário: Aqui se é possível escolher se a pessoa é física ou jurídica. Se a opção selecionada for física, são inseridos um CPF, um nome civil e um nome social. Caso seja jurídica, deverão ser colocados um CNPJ, um nome fantasia e uma razão social. Também são colocadas as informações de contato e as informações residenciais do empresário.



Imagen 10 – Gerenciamento de empresário

Gerenciamento de Empresários: Onde são consultados os empresários cadastrados no sistema. É possível fazer a pesquisa por filtros ou manualmente. Também é possível ativar ou reativar um empresário, consultar ou alterar suas informações e até mesmo excluir um.

Gerenciamento de Artista					
Digite o que deseja buscar				Procurar	<input type="radio"/> Ativos <input checked="" type="radio"/> Todos <input type="radio"/> Desativados
Cód	Nome	Tipo Artista	Sexo	Facebook	Twitter
2	BARBIXAS	ARTISTA	MASCULINO	osBARBIXAS	
9	DANILLO GENTILI	ARTISTA	MASCULINO	ComeduHouse	ComeduHou
6	DIEGO	ARTISTA-FIXO	MASCULINO	FabioPorchat	FabiOFICIA
1	FABIO PORCHAT	ARTISTA	MASCULINO	MarcLua	LUQUEMAR
4	MARCO LUQUE	ARTISTA	MASCULINO	TataOFICIAL	Werneck
3	TATA WERNECK	ARTISTA	FEMININO		
5	TOM CAVALCANTE	ARTISTA	MASCULINO		
7	WHINDERSSON	ARTISTA	MASCULINO	whinderssonOFICIAL	

CTRL + F Excluir Desativar Reativar Consultar Alterar

Imagen 11 – Gerenciamento de artistas

Gerenciamento de Artistas: Onde são consultados os artistas cadastrados no sistema. É possível fazer a pesquisa por filtros ou manualmente. Também é possível ativar ou reativar um artista, consultar ou alterar suas informações e até mesmo excluir um.



Imagen 12 – Gerenciamento de evento

Gerenciamento de Eventos: Onde são consultados os eventos cadastrados no sistema. É possível fazer a pesquisa por filtros ou manualmente. Também é possível ativar ou reativar um evento, consultar ou alterar suas informações e até mesmo excluir um. Em vermelho estão mostrados eventos já realizados.

Ocorrências (Log)		
<input checked="" type="radio"/> Alterados <input type="radio"/> Inseridos	<input checked="" type="radio"/> Excluídos <input type="radio"/> Logon/Logoff	27/07/2018 á 27/07/2018
Setor <input type="button" value="Todos"/>		Entidade Especifica(s) <input type="text"/> <input type="button" value="Procurar"/>
Data da Ocorrência	Tipo de Ocorrência	Responsável
27/07/2018 14:22	LOGON	CLAUDIO OLIVEIRA
27/07/2018 14:02	LOGOFF	CLAUDIO OLIVEIRA
27/07/2018 14:02	LOGON	CLAUDIO OLIVEIRA
27/07/2018 14:02	LOGOFF	CLAUDIO OLIVEIRA
27/07/2018 13:56	LOGON	CLAUDIO OLIVEIRA
27/07/2018 13:56	LOGOFF	CLAUDIO OLIVEIRA
27/07/2018 13:56	LOGON	CLAUDIO OLIVEIRA
27/07/2018 13:54	LOGON	CLAUDIA SOUZA
27/07/2018 13:53	LOGOFF	CLAUDIA SOUZA
27/07/2018 13:52	LOGON	CLAUDIA SOUZA
27/07/2018 13:52	LOGOFF	SHIRLEY NASCIMENTO
27/07/2018 13:52	LOGON	SHIRLEY NASCIMENTO
27/07/2018 13:50	LOGOFF	CLAUDIO OLIVEIRA
27/07/2018 13:49	LOGON	CLAUDIO OLIVEIRA

Imagen 13 - Log

Log: São as ocorrências do sistema, tais como: a criação de um novo artista; a alteração de um funcionário, etc. Tudo que foi feito é registrado aqui.



Imagen 14 – Gerenciamento de Usuário

Gerenciamento de Usuários: Onde são consultados os usuários cadastrados no sistema, que pode ser um cliente ou funcionário da casa. É possível fazer a pesquisa por filtros ou manualmente. Também é possível consultar ou alterar as informações de login dos mesmos.



Imagen 15 - Novidades

Novidades: Nessa tela serão escolhidas as imagens que serão usadas como novidades no banner rotativo do site. É possível adicionar ou retirar imagens. A posição de cada uma é representada na linha abaixo da imagem.



Imagen 16 – Formas de Pagamento

Formas de Pagamento: Onde são mostradas todas as formas de pagamento que um cliente pode escolher para realizar um pagamento. É possível adicionar novas formas de pagamento, como também alterá-las ou excluí-las.



Imagen 17 – Funções

Funções: Serve para cadastrar, alterar ou excluir uma função do sistema. Para o cadastro deverá ser escolhido um nome para a função, verificar se a função terá um login próprio e um nível de acesso.



Imagen 18 – Pagamento de ingresso

Pagamento de Ingresso: Após os ingressos serem escolhidos na tela de venda de ingressos, o processo é continuado nessa tela. Aqui será mostrado o valor total dos ingressos, adição de descontos caso haja um código promocional para ser usado e seu respectivo percentual, um troco se houver, também deverá escolhida uma forma de pagamento para a compra.

Cadastro de Contas
*Todos os campos com * são obrigatórios*

Ativo

• Conta para Receber **• Conta para Pagar**

Data de Lançamento 27/07/2018	Departamento	Formas de Pagamento
Situação	* Descrição	Dinheiro
* Data de Vencimento 27/07/2018	* Valor Total R\$0	* Valor
* Data Entrega 27/07/2018		<input type="button" value="Adicionar"/> <input type="button" value="Excluir"/>

Imagen 19 – Cadastro de contas

Cadastro de Contas: Aqui será cadastrada uma conta para pagar ou receber, deverá ser informado a descrição da conta, o departamento, data de lançamento e de vencimento, situação, a forma de pagamento e o valor pago e nominal.

Gerenciamento de Contas 			
Digite o que deseja buscar		04/08/2017 à 11/09/2018	<input type="button" value="Procurar"/> <input checked="" type="radio"/> Alivos <input type="radio"/> Todos <input type="radio"/> Desativados
 FILTRO			
Código da Conta	Tipo da Conta	Descrição	
7	Receber	ALUGUEL DO BURGUER KING	
5	Pagar	CONTA DE LUZ	
6	Receber	INGRESSO DE 1 ASSENTO(S), VENDIDO POR: CLAUDIA SOUZA	
14	Receber	INGRESSO DE 1 ASSENTO(S), VENDIDO POR: CLAUDIA SOUZA	
18	Receber	INGRESSO DE 1 ASSENTO(S), VENDIDO POR: CLAUDIO OLIVEIRA	
1	Receber	INGRESSO DE 1 ASSENTO(S), VENDIDO POR: CLAUDIO OLIVEIRA	
13	Receber	INGRESSO DE 1 ASSENTO(S), VENDIDO POR: CLAUDIO OLIVEIRA	
11	Receber	INGRESSO DE 1 ASSENTO(S), VENDIDO POR: CLAUDIO OLIVEIRA	
21	Receber	INGRESSO DE 1 ASSENTO(S), VENDIDO POR: CLAUDIO OLIVEIRA	
12	Receber	INGRESSO DE 1 ASSENTO(S), VENDIDO POR: CLAUDIO OLIVEIRA	
10	Receber	INGRESSO DE 2 ASSENTO(S), VENDIDO POR: CLAUDIA SOUZA	

Imagem 20 – Gerenciamento de contas

Gerenciamento de Contas: Onde são consultadas as contas cadastradas no sistema, é possível fazer a pesquisa por filtros ou uma pesquisa manual. Também é possível ativar ou reativar uma conta, consultar ou alterar suas informações e excluir uma conta. As contas em vermelho podem ser contas já pagas ou que passaram do prazo para se pagar.

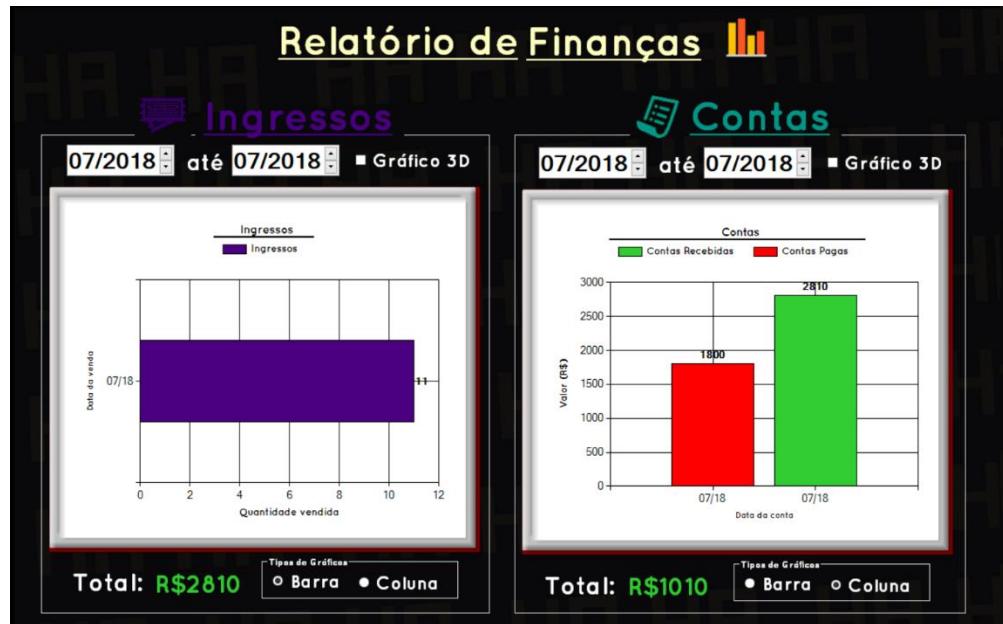


Imagen 21 – Relatório de finanças

Relatório de Finanças: Aqui são mostrados os relatórios das vendas dos ingressos em um mês pelo gráfico que pode ser em barras ou colunas e a receita total recebida pela casa com as vendas. Também é mostrado o relatório de contas a receber ou a pagar em um mês. As contas a receber serão subtraídas pelas contas a pagar e então será gerado o lucro no mês.

Imagen 22 – Cadastro de funcionários

Cadastro de Funcionários: Aqui é onde serão colocadas as informações da pessoa, o contato e o local, para o funcionário também deverá ser atribuído uma função, dependendo da função esse funcionário terá um login para acessar o sistema.

Cód Funcionário	Nome	Sexo	Data de Nascimento	Ativo	Função
4	CLAUDIA SOUZA	FEMININO	15/07/1990	1	Gerente
1	CLAUDIO OLIVEIRA	MASCULINO	11/09/1985	1	Gerente
3	JOSÉ ROBERTO	MASCULINO	07/10/1985	1	Segurança
2	SHIRLEY NASCIMENTO	FEMININO	15/07/1980	1	Atendente

Imagen 23 – Gerenciamento de funcionários

Gerenciamento de Funcionários: Aqui serão mostrados os funcionários registrados no sistema onde também pode ser feito alterações nos funcionários.



Imagen 24 – Loading

Loading: Tela de carregamento do sistema após o login ser efetuado, antes da iniciação do Hub.



Imagen 25 - Descontos

Descontos: Esta tela serve para adicionarmos um cupom de desconto no sistema, que poderá ser usado na venda de um ingresso. Deve ser colocada uma porcentagem, e o nome do código.

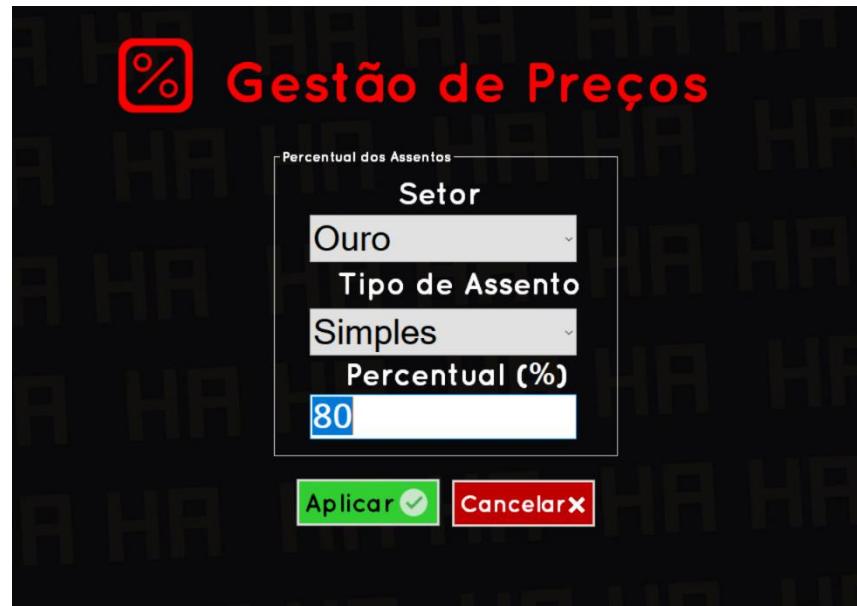


Imagen 26 – Gestão de Preços

Gestão de Preços: Aqui serão escolhidos os valores que serão usados como acréscimo na venda de um assento. Por exemplo, ao escolher um assento ouro o cliente terá que pagar pelo valor base que será o do ingresso mais esse acréscimo. O valor é atribuído em porcentagem.



Imagen 27 – Consulta de Ingresso

Consulta de Ingresso: Aqui poderá ser feita a consulta de um ingresso vendido e ver as respectivas informações que foram escolhidas na venda. Também é possível imprimir o ingresso e o recibo.



Imagen 28 – Consulta de Ingressos - Assentos

Consulta de Ingressos - Assentos: Aqui são mostradas as informações dos assentos comprados.



Imagen 29 – Consulta de Ingressos

Consulta de Ingressos - Evento: Aqui são mostradas informações do evento e seus respectivos artistas.



Imagen 30 – Consulta de Ingressos - Cliente

Consulta de Ingressos - Cliente: Aqui são mostradas informações do cliente que efetuou a compra do ingresso, também com a opção de consultar suas informações por completo.

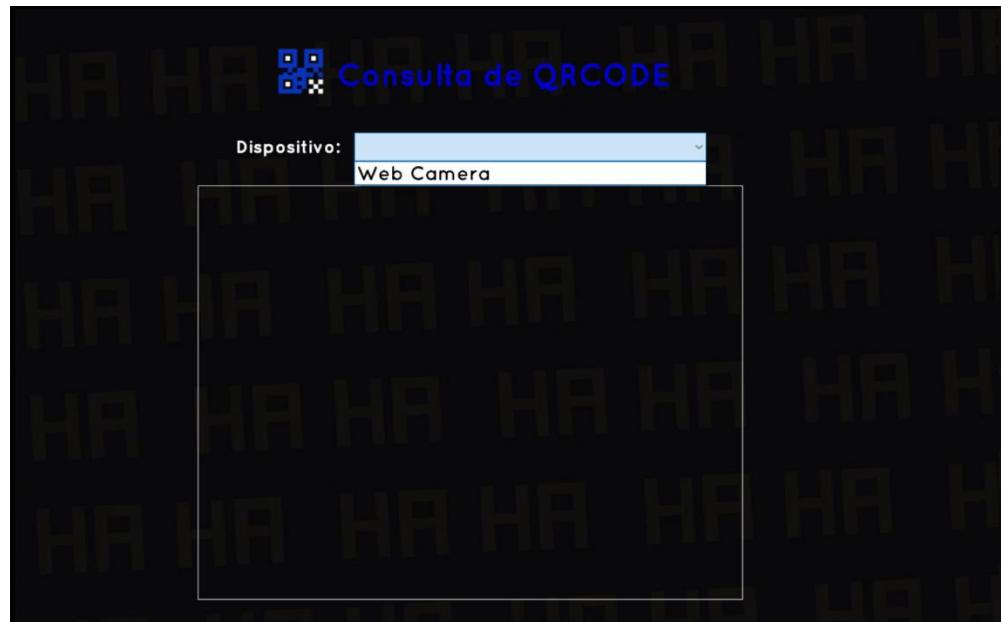


Imagen 31 – Consulta de QRCode

Consulta de QRCode: Um outro método de consultar a venda de um ingresso é utilizando um QRCode que foi gerado para esse ingresso. Utilizando a câmera, é só mostrar o código e você será redirecionado para o ingresso vendido.



Imagen 32 – Reembolso de Ingresso

Reembolso de Ingresso: Nessa tela é possível fazer o reembolso de um ingresso vendido, onde é mostrado o responsável que está logado, e a data de cancelamento. E então deverá ser escrito o motivo pelo qual o ingresso está sendo reembolsado.

Gerenciamento de Ingressos							
Digite o que deseja buscar				04/08/2017	à	12/08/2018	<input type="button" value="Procurar"/> <input checked="" type="radio"/> Pagos <input type="radio"/> A pagar <input type="radio"/> Todos <input type="radio"/> Cancelados
<input type="button" value="FILTRAR"/>							
Cód Venda	Data da Venda	Situação	Preço Total	Data do Evento	Horário do Evento	Ti	
21	30/07/2018 11:02	Pago	98	31/07/2018	18:00 até 20:30	D	
17	30/07/2018 11:00	Cancelado	98	31/07/2018	18:00 até 20:30	D	
16	30/07/2018 08:41	Cancelado	483	03/08/2018	18:00 até 19:00		
15	28/07/2018 19:04	Pago	100	01/08/2018	18:00 até 19:30	TERI	
14	28/07/2018 18:30	Pago	360	01/08/2018	18:00 até 19:30	TERI	
13	26/07/2018 22:25	Pago	360	01/08/2018	18:00 até 19:30	TERI	
12	26/07/2018 22:22	Pago	360	01/08/2018	18:00 até 19:30	TERI	
11	26/07/2018 22:21	Pago	180	01/08/2018	18:00 até 19:30	TERI	
10	25/07/2018 22:30	Pago	200	01/08/2018	18:00 até 19:30	TERI	
9	25/07/2018 22:28	Pago	200	01/08/2018	18:00 até 19:30	TERI	
8	25/07/2018 17:31	Pago	100	01/08/2018	18:00 até 19:30	TERI	

Imagen 33 – Gerenciamento de Ingressos

Gerenciamento de Ingressos: Aqui é onde são mostrados todos os ingressos vendidos e suas informações. Também é possível consultar a venda ou reembolsar um ingresso.

Cadastro de Cliente	
<i>Todos os campos com * são obrigatórios</i>	
<input type="checkbox"/> Ativo	
Nome * <input type="text"/>	Data de Nascimento <input type="text"/> 27/07/2000
Sexo <input type="text"/> N.I	CPF * <input type="text"/> <input checked="" type="checkbox"/>
Email <input type="text"/> <input checked="" type="checkbox"/>	
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> LOGIN ? Usuário * <input type="text"/> Senha * <input type="password"/> </div> <div style="display: flex; justify-content: space-around;"> Pergunta Secreta * Resposta * </div>	
<input type="button" value="Limpar"/> <input type="button" value="Salvar"/> <input type="button" value="Cancelar"/>	

Imagen 34 – Cadastro de Clientes

Cadastro de Cliente: Nessa tela será cadastrado um cliente no sistema, deverão ser colocadas as informações básicas e então é feito um login para o cliente acessar o site e o mobile.

Consulta de Cliente

Todos os campos com * são obrigatórios

Nome *	Data de Nascimento
ARTHUR PORTO	01/04/1999
Sexo	CPF *
Masculino	194495780-44 <input checked="" type="checkbox"/>
Email	
arthur@gmail.com	<input checked="" type="checkbox"/>
LOGIN Usuário * 19449578044 Senha * ****	
Pergunta Secreta * COD STEAM Resposta * ****	
<input type="button" value="Consultar"/>	
<input type="button" value="Cancelar"/>	

Histórico

• Dia 25/07/2018 22:30

20/07/2018 à 27/07/2018

Imagen 35 – Consulta de Clientes

Consulta de Cliente: Aqui é possível verificar as informações do cliente e o login, como também ver o histórico de ingressos comprados por ele.

Gerenciamento de Cliente

Cód	Nome	Sexo	Data de Nascimento	Ativo	CPF
1	ARTHUR PORTO	Masculino	01/04/1999	1	194495780-44
4	FABIO COSTA	Masculino	24/07/1987	1	812918300-57
3	FERNANDO LUIS	Masculino	11/05/1989	1	525735820-43
2	WILSON SOARES DE ABREU	Masculino	15/02/1975	1	832633900-82

CTRL + F

Excluir Desativar Reativar Consultar Alterar

Imagen 36 – Gerenciamento de Clientes

Gerenciamento de Cliente: Aqui são gerenciados todos os clientes cadastrados no sistema, é possível excluir, desativar ou ativar, consultar e alterar um cliente. É possível procurar por um cliente específico na barra ou utilizar o filtro.

3 SISTEMA WEB DINÂMICO (ASP.Net)

Nome	Função
Arthur Porto	Programador Full stack e analista
Diego Gomes	Documentação e analista
Gabriel Leandro	Programador Front – End e analista
Lucas Camargo	Programador Back – End e banco de dados
Marcos Vinicius	Banco de dados e analista

Tabela 1 – Funções dos participantes

3.1 Gerenciamento de Projeto

3.1.1 Termo de Abertura do Projeto (TAP)

1. DADOS GERAIS

Diretor Comercial	Telefone
Glauber Santos Lima	11 2584-2712
Diretor de Desenvolvimento	Telefone
Fernando Luiz	11 3227-7315

2. IDENTIFICAÇÃO DO CLIENTE

Cliente	Nome do Principal Contato no Cliente	Telefone
Comedy House LTDA	Cláudio Oliveira	11 2235-9706
Endereço	Complemento	
Rua Augusta, 1100		
Cidade	UF	
São Paulo	SP	

Nome do responsável pela empresa	E-mail	Telefone
Cláudio Oliveira	comedyhouse.oficial@gmail.com	11 2577-9956

3. DADOS DA PROPOSTA

Valor Proposta	Nº da Proposta	Data da Proposta
R\$ 25 000,00	03	17/02/2017

3.1.2 Estrutura Analítica do Projeto (EAP)

3.1.3 Cronograma

3.2 Escopo

3.2.1 Objetivo do Documento

3.2.1.1 Para o Cliente

Para o cliente, a documentação é o que comprova a devida realização do projeto. Com este documento, o cliente poderá analisar e verificar se todas as exigências feitas para cada parte do sistema foram implantadas corretamente, saberá se algo estiver errado ou faltando e se o trabalho foi feito corretamente, porque tudo que foi requisitado está guardado aqui.

3.2.1.2 Para o Projetista

Para a equipe, a documentação é o que mostra tudo que foi solicitado pelo cliente desde o início do projeto. Com ela nós temos todas as informações do sistema e tudo que foi feito baseado nas exigências do cliente. Sendo assim, caso o cliente faça reclamações após o término do projeto dizendo que algo não foi feito como ele queria ou algo que deixamos de fazer, temos a documentação para a devida comprovação de realização.

3.3 Descrição do Ambiente

3.3.1 Descrição do Processo / Problema

Devido à maior demanda na busca por ingressos e a indisponibilidade online, a casa de shows necessitou da criação de um Website para a venda dos mesmos. Além disso, a página também possuirá outros recursos, como: um setor exclusivo para suporte ao usuário, uma opção de cadastro para quem não possuir uma conta e um perfil para quem já possuí-la, um setor apresentando os artistas associados à casa e um setor apresentando os eventos futuros.

Na página principal, o usuário geral (aquele que pode estar logado ou não) tem acesso ao Banner Rotativo, que conterá as novidades da casa (que serão adicionadas pelo sistema Desktop).

Na página de Ajuda, que é o suporte, o usuário geral tem acesso às perguntas mais frequentes feitas por outros usuários, as informações de contato da casa, como número de celular, e-mail, etc. e poderá ver os feedbacks públicos enviados. Porém, apenas quem possuir login terá direito a publicar seu feedback.

Na página dos Artistas, serão listados todos os artistas que já se apresentaram na Comedy House e todos terão acesso à ela. Porém, apenas quem possuir uma conta terá o direito de ver os detalhes de cada artista, como suas redes sociais, informações pessoais, etc.

No setor dos Eventos, assim como na página dos Artistas, serão listados todos os Eventos disponíveis para a compra de ingressos. Assim que selecionado o evento, o usuário é direcionado para uma página com mais informações específicas daquele evento. Caso prossiga a compra, é requisitado que o usuário faça login (caso ainda não tenha o feito) e é direcionado para a página dos Ingressos.

Nos Ingressos, são escolhidos os assentos desejados pelo usuário e ele é direcionado para o pagamento.

Na página do Pagamento, o usuário pode inserir um voucher de desconto (caso possua um), selecionar uma ou várias formas de pagamento e a quantidade de parcelas ideais e assim finalizar a compra.

3.3.2 Descrição do Ambiente / Infra-estrutura

A Comedy House fica em um edifício de 1 andar (similar a um Teatro) e no total possui 1000 m², sendo constituído por:

- 6 Banheiros (feminino, masculino e deficiente) de 25 m² cada;
- 1 Lanchonete de 150 m²;
- 1 Palco de 36 m²;
- Um espaço de 594 m² para todos os assentos que são divididos em tipos (Duplo, Especial, Simples) categorias (Ouro, Prata, Bronze)
 - Ouro: Assentos mais próximos ao palco;
 - Prata: Assentos não tão próximos ao palco;
 - Bronze: Assentos distantes do palco;
- Recepção – Um espaço de 50 m² de recepção, lugar onde ficam os computadores, que são 3 no total;
 - Camarim – Um espaço de 20 m² de camarim; O edifício tem a capacidade máxima de 170 espectadores e possui 20 funcionários, sendo 13 faxineiros (as), 3 recepcionistas, 4 seguranças.

3.3.3 Metas e Objetivos do Sistema

A principal meta do website é suportar a grande demanda por ingressos pelos clientes, que antes, para adquiri-los, necessitavam ir até o estabelecimento e efetuar a compra pessoalmente.

3.3.4 Sistemas atualmente em uso.

Em relação ao website, não havia um sistema em uso. A casa não possuía um website para a venda de ingressos similar ao que ela nos solicitou

3.3.5 Escopo do Sistema

O escopo do sistema se baseia na venda do ingresso, pois ele é a característica principal do sistema, uma vez que a casa de show tem como principal fonte de renda os ingressos vendidos nos eventos, de modo que as pessoas vão comprá-los baseados nos setores que são lhes proporcionados. Por esta razão os preços são acessíveis, estando de acordo com a quantia no qual os clientes estão aptos a pagar, aumentando assim, a lucratividade do negócio e facilitando a venda dos mesmos. Os ingressos são simples e possuem as informações necessárias que o cliente precisa saber, como o evento, horário, local e os artistas. O software em que o sistema da venda do ingresso é usado e simples e objetivo, onde é seguido uma ordem lógica para que o ingresso seja gerado.

3.3.6 Funções do Sistema

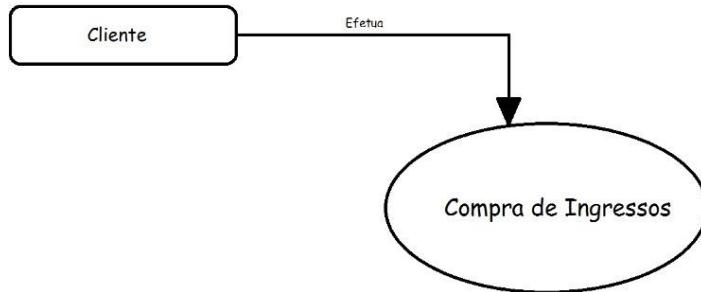
A principal função exercida pelo Web é a venda dos ingressos para os usuários que possuem um cadastro no sistema. Essa venda é feita para qualquer evento disponível e o usuário pode escolher seus determinados assentos e quantas formas de pagamento achar ideal.

3.3.7 Regras de Negócio

No site, a regra de negócios será controlada pela linguagem de programação ASP.NET, e o banco de dados pelo SQL Server 2017.

3.3.8 DFD – Diagrama de Fluxo de Dados

3.3.9 DFD Nível 0 - Diagrama de Contexto



3.3.10 DFD Nível 1

3.4 Dados do Sistema

3.4.1 Dados Armazenados

3.4.2 Dicionário de Termo de Dados

<u>NOME DO ATRIBUTO</u>	<u>TIPO DO ATRIBUTO</u>	<u>DESCRÍÇÃO DO ATRIBUTO</u>
NOME CAMPO	TIPO, AUTO NUM., PK	Descrição do Campo.
NOME CAMPO	TIPO, FK	Descrição do Campo.

Tabela 2 – Dicionário de dados

3.4.3 Diagrama de Entidade e Relacionamento (DER)

3.4.4 Script de criação do Banco de Dados (SQL)

DADOS

----- DELETAMENTO DE DADOS

```

DELETE FROM TB_AGENTE;
DELETE FROM TB_ARTISTA;
DELETE FROM TB_ARTISTA_FIXO;
DELETE FROM TB_ARTISTA_GERAL;
DELETE FROM TB_ASSENTO_CLI;
DELETE FROM TB_BANNER;
DELETE FROM TB_CANCELAR_VENDA;
DELETE FROM TB_CLIENTE;
DELETE FROM TB_CONTAS;
DELETE FROM TB_DATA_CONTA;
DELETE FROM TB_DESCONTOS;
DELETE FROM TB_DESCONTOS_CLIENTE;
DELETE FROM TB_DETALHE_CONTA;
DELETE FROM TB_DETALHE_CONTA_VENDA;
DELETE FROM TB_ENDERECO;
DELETE FROM TB_EVENTO;
DELETE FROM TB_EVENTO_ARTISTA;
DELETE FROM TB_EVENTO_INGRESSO;
DELETE FROM TB_FORMA_PAGAMENTO;
DELETE FROM TB_FUNCAO;
DELETE FROM TB_FUNCIONARIO;
DELETE FROM TB_ITEM_VENDA;
DELETE FROM TB_LOG;
DELETE FROM TB_LOG_ALTERACAO;
DELETE FROM TB_LOG_EXCLUSAO;
DELETE FROM TB_LOG_INSERCAO;
DELETE FROM TB_LOGIN;
DELETE FROM TB_NIVELACESSO;

```

```

DELETE FROM TB_PAGAMENTO;
DELETE FROM TB_PERGUNTASECRETA;
DELETE FROM TB_PESSOA;
DELETE FROM TB_PRECOS;
DELETE FROM TB_REDESOCIAL;
DELETE FROM TB_VENDA_INGRESSO;

```

-----TIPOS DE PESSOA

```

USE TCCSHOW
GO
INSERT INTO TB_TIPO_PESSOA(ID_TIPO_PESSOA, TIPO_PESSOA)
VALUES(1,'FUNCIONARIO');
INSERT INTO TB_TIPO_PESSOA(ID_TIPO_PESSOA, TIPO_PESSOA)
VALUES(2,'CLIENTE');
INSERT INTO TB_TIPO_PESSOA(ID_TIPO_PESSOA, TIPO_PESSOA)
VALUES(3,'ARTISTA');
INSERT INTO TB_TIPO_PESSOA(ID_TIPO_PESSOA, TIPO_PESSOA)
VALUES(4,'ARTISTA-FIXO');

```

-----FORMAS DE PAGAMENTO

```

INSERT INTO TB_FORMA_PAGAMENTO(ID_FORMA_PAGAMENTO, FORMA_PAGAMENTO)
VALUES(1,'DINHEIRO');
INSERT INTO TB_FORMA_PAGAMENTO(ID_FORMA_PAGAMENTO, FORMA_PAGAMENTO)
VALUES(2,'CARTAO DE CREDITO');
INSERT INTO TB_FORMA_PAGAMENTO(ID_FORMA_PAGAMENTO, FORMA_PAGAMENTO)
VALUES(3,'CARTAO DE DEBITO');
INSERT INTO TB_FORMA_PAGAMENTO(ID_FORMA_PAGAMENTO, FORMA_PAGAMENTO)
VALUES(4,'BOLETO');
INSERT INTO TB_FORMA_PAGAMENTO(ID_FORMA_PAGAMENTO, FORMA_PAGAMENTO)
VALUES(5,'CHEQUE');

```

-----FUNÇÕES

```

INSERT INTO TB_FUNCAO(ID_FUNCAO, FUNCAO)
VALUES(1,'GERENTE');
INSERT INTO TB_FUNCAO(ID_FUNCAO, FUNCAO)
VALUES(2,'ATENDENTE');
INSERT INTO TB_FUNCAO(ID_FUNCAO, FUNCAO)
VALUES(3,'AUXILIAR DE LIMPEZA');
INSERT INTO TB_FUNCAO(ID_FUNCAO, FUNCAO)
VALUES(4,'SEGURANCA');

```

-----NIVEIS DE ACESSO

```

INSERT INTO TB_NIVELACESSO(ID_NIVELACESSO, NIVELACESSO)
VALUES(1,'ADMINISTRADOR');
INSERT INTO TB_NIVELACESSO(ID_NIVELACESSO, NIVELACESSO)

```

```

VALUES(2,'FUNCIONARIO COMUM');
INSERT INTO TB_NIVELACESSO(ID_NIVELACESSO, NIVELACESSO)
VALUES(3,'USUARIO CLIENTE');

-----GERENTE CLAUDIO
INSERT INTO TB_PESSOA(ID_PESSOA, NOME, DATA_NASC, SEXO, ATIVO,
ID_TIPO_PESSOA, DATA_CRIACAO)
VALUES(1,'CLAUDIO','11/09/1985','MASCULINO',1,1,'01/01/2017');
INSERT INTO TB_ENDERECHO(ID_ENDERECHO, CEP, NUMERO, COMPLEMENTO)
VALUES(1,'06413040','120',NULL);
INSERT INTO TB_FUNCIONARIO(ID_FUNC, ID_FUNCAO, ID_ENDERECHO, CPF,
EMAIL, TELEFONE, ID_PESSOA)
VALUES(1,1,1,'503648518-13',NULL,'(12) 3456-7891',1);
INSERT INTO TB_PERGUNTASECRETA(ID_PERGUNTASECRETA, DESCRICAO,
RESPOSTA)
VALUES(1, ' 1+2+3 ','123');
INSERT INTO
TB_LOGIN(ID_LOGIN, ID_NIVELACESSO, ID_PERGUNTASECRETA, USUARIO, SENHA,
DATAULTIMOACESSO, ID_PESSOA)
VALUES(1, 1,1,'GERENTE','123',GETDATE(),1);

-----TIPO DE ASSENTO

INSERT INTO TB_TIPO_ASSENTO (ID_TIPO_ASSENTO, TIPO_ASSENTO)
VALUES(1,'SIMPLES');
INSERT INTO TB_TIPO_ASSENTO (ID_TIPO_ASSENTO, TIPO_ASSENTO)
VALUES(2,'DUPLO');
INSERT INTO TB_TIPO_ASSENTO (ID_TIPO_ASSENTO, TIPO_ASSENTO)
VALUES(3,'ESPECIAL');

-----TIPO DE LOG

INSERT INTO TB_TIPO_LOG (ID_TIPO_LOG, TIPO_LOG)
VALUES(1,'ALTERAÇÃO');
INSERT INTO TB_TIPO_LOG (ID_TIPO_LOG, TIPO_LOG)
VALUES(2,'EXCLUSÃO');
INSERT INTO TB_TIPO_LOG (ID_TIPO_LOG, TIPO_LOG)
VALUES(3,'INSERÇÃO');
INSERT INTO TB_TIPO_LOG (ID_TIPO_LOG, TIPO_LOG)
VALUES(4,'LOGON');
INSERT INTO TB_TIPO_LOG (ID_TIPO_LOG, TIPO_LOG)
VALUES(5,'LOGOFF');

-----TABELAS LOG

INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)
VALUES(1,'ARTISTAS');
INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)
VALUES(2,'CLIENTES');
INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)
VALUES(3,'CONTAS');
INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)
VALUES(4,'DESCONTOS');
INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)
VALUES(5,'EMPRESÁRIOS');
INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)

```

```

VALUES(6,'EVENTOS');
INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)
VALUES(7,'FORMAS PAGS');
INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)
VALUES(8,'FUNÇÕES');
INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)
VALUES(9,'FUNCIONÁRIOS');
INSERT INTO TB_TABELAS_LOG (ID_TABELA_LOG, TABELA)
VALUES(10,'NOVIDADES');

-----SETOR

INSERT INTO TB_SETOR (ID_SETOR, SETOR)
VALUES(1,'OURO');
INSERT INTO TB_SETOR (ID_SETOR, SETOR)
VALUES(2,'PRATA');
INSERT INTO TB_SETOR (ID_SETOR, SETOR)
VALUES(3,'BRONZE');

-----NAO IDENTIFICADO

INSERT INTO TB_PESSOA (ID_PESSOA, NOME, DATA_NASC, SEXO, ATIVO,
ID_TIPO_PESSOA, DATA_CRIACAO)
VALUES(0,'NAO IDENTIFICADO',NULL, NULL, 1, 2, NULL);
INSERT INTO TB_CLIENTE(ID_CLIENTE, CPF, ID_PESSOA, EMAIL)
VALUES(0,NULL,0,NULL);

-----PREÇOS

INSERT INTO TB_PRECOS(ID_SETOR, ID_TIPO_ASSENTO, PERC)
VALUES(1,1,80);
INSERT INTO TB_PRECOS(ID_SETOR, ID_TIPO_ASSENTO, PERC)
VALUES(1,2,100);
INSERT INTO TB_PRECOS(ID_SETOR, ID_TIPO_ASSENTO, PERC)
VALUES(1,3,0);
INSERT INTO TB_PRECOS(ID_SETOR, ID_TIPO_ASSENTO, PERC)
VALUES(2,1,30);
INSERT INTO TB_PRECOS(ID_SETOR, ID_TIPO_ASSENTO, PERC)
VALUES(2,3,0);
INSERT INTO TB_PRECOS(ID_SETOR, ID_TIPO_ASSENTO, PERC)
VALUES(3,1,0);

DELETE FROM TB_ASSENTO;
-----ASSENTOS

INSERT INTO TB_ASSENTO (ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 12, 'A', 3, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 13, 'A', 3, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 14, 'A', 3, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)

```

```
VALUES ( 15, 'A', 3, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 16, 'A', 3, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 17, 'A', 3, 1);

INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 1, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 2, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 3, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 4, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 5, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 6, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 7, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 8, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 9, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 10, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 11, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 12, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 13, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 14, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 15, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 16, 'B', 1, 1);
```

```
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 17, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 18, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 19, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 20, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 21, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 22, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 23, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 24, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 25, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 26, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 27, 'B', 1, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 28, 'B', 1, 1);

INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '1e2', 'C', 2, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '4e5', 'C', 2, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '7e8', 'C', 2, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '10e11', 'C', 2, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '13e14', 'C', 2, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '15e16', 'C', 2, 1);
```

```
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '18e19', 'C', 2, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '21e22', 'C', 2, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '24e25', 'C', 2, 1);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( '27e28', 'C', 2, 1);

INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 1, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 2, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 3, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 4, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 5, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 6, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 7, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 8, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 9, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 10, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 11, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 12, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 13, 'D', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 14, 'D', 1, 2);
```



```
VALUES ( 22, 'E', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 23, 'E', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 24, 'E', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 25, 'E', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 26, 'E', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 27, 'E', 1, 2);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 28, 'E', 1, 2);

INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 1, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 2, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 3, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 4, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 5, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 6, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 7, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 8, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 9, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 10, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 11, 'F', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 12, 'F', 1, 3);
```



```
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 3, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 4, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 5, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 6, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 7, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 8, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 9, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 10, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 11, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 12, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 13, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 14, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 15, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 16, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 17, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 18, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 19, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 20, 'G', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
```



```

INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 19, 'H', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 20, 'H', 1, 3);
INSERT INTO TB_ASSENTO ( ASSENTO_NUMER, ASSENTO_FILEIRA,
ID_TIPO_ASSENTO, ID_SETOR)
VALUES ( 21, 'H', 1, 3);
-----
```

TABELAS

```

--DROP DATABASE TCCSHOW;

CREATE DATABASE TCCSHOW;

USE TCCSHOW
GO
--DROP TABLE TB_TIPO_PESSOA;
CREATE TABLE TB_TIPO_PESSOA(
    ID_TIPO_PESSOA INTEGER NOT NULL,
    TIPO_PESSOA VARCHAR(12) NOT NULL,
    CONSTRAINT TB_TIPO_PESSOA_PK PRIMARY KEY (ID_TIPO_PESSOA)
);

--DROP TABLE TB_PESSOA;
CREATE TABLE TB_PESSOA(
    ID_PESSOA INTEGER NOT NULL,
    NOME VARCHAR(150) NOT NULL,
    DATA_NASC DATE,
    SEXO VARCHAR(9),
    ATIVO NUMERIC(1) NOT NULL,
    ID_TIPO_PESSOA INTEGER NOT NULL,
    DATA_CRIACAO DATE,
    CONSTRAINT TB_PESSOA_PK PRIMARY KEY (ID_PESSOA),
    CONSTRAINT TB_TIPO_PESSOA_FK FOREIGN KEY(ID_TIPO_PESSOA)
    REFERENCES TB_TIPO_PESSOA(ID_TIPO_PESSOA)
);

--DROP TABLE TB_CEP;
CREATE TABLE TB_CEP(
    CEP VARCHAR(8) NOT NULL,
    UF VARCHAR(2),
    DESCRICAO VARCHAR(256),
    CIDADE VARCHAR(27),
    BAIRRO VARCHAR(128),
    CONSTRAINT TB_CEP_PK PRIMARY KEY (CEP)
);
```

```

--DROP TABLE TB_ENDERECO;
CREATE TABLE TB_ENDERECO(
    ID_ENDERECO INTEGER NOT NULL,
    CEP VARCHAR(8),
    NUMERO NUMERIC(6),
    COMPLEMENTO VARCHAR(120),

    CONSTRAINT TB_ENDERECO_PK PRIMARY KEY (ID_ENDERECO),
    CONSTRAINT TB_CEP_FK FOREIGN KEY(CEP)
    REFERENCES TB_CEP(CEP)
);

--DROP TABLE TB_AGENTE;
CREATE TABLE TB_AGENTE(
    ID_AGENTE INTEGER NOT NULL,
    ID_ENDERECO INTEGER,
    TELEFONE VARCHAR(14),
    TIPO_PESSOA VARCHAR(8) NOT NULL,
    DOCUMENTO VARCHAR(18),
    EMAIL VARCHAR(120),
    NOME_PRINCIPAL VARCHAR(120),
    NOME_SECUNDARIO VARCHAR(120),
    ATIVO NUMERIC(1),
    DATA_CRIACAO DATE,

    CONSTRAINT TB_AGENTE_PK PRIMARY KEY (ID_AGENTE),
    CONSTRAINT TB_AGENTE_ENDERECO_FK FOREIGN KEY(ID_ENDERECO)
    REFERENCES TB_ENDERECO(ID_ENDERECO)
);

--DROP TABLE TB_REDESOCIAL;
CREATE TABLE TB_REDESOCIAL(
    ID_REDESOCIAL INTEGER NOT NULL,
    FACEBOOK VARCHAR(120),
    TWITTER VARCHAR(120),
    INSTAGRAM VARCHAR(120),

    CONSTRAINT TB_REDESOCIAL_PK PRIMARY KEY (ID_REDESOCIAL)
);

--DROP TABLE TB_ARTISTA_GERAL;
CREATE TABLE TB_ARTISTA_GERAL(
    ID_ARTISTA_GERAL INTEGER NOT NULL,
    ID_REDESOCIAL INTEGER,
    ID_PESSOA INTEGER NOT NULL,
    URL_IMG VARCHAR(250),

    CONSTRAINT TB_ART_PESSOA_FK FOREIGN KEY (ID_PESSOA)
    REFERENCES TB_PESSOA(ID_PESSOA),
    CONSTRAINT TB_REDESOCIAL_FK FOREIGN KEY (ID_REDESOCIAL)
    REFERENCES TB_REDESOCIAL(ID_REDESOCIAL),
    CONSTRAINT TB_ARTISTA_GERAL_PK PRIMARY KEY (ID_ARTISTA_GERAL)
);

--DROP TABLE TB_ARTISTA;
CREATE TABLE TB_ARTISTA(

```

```

ID_AGENTE INTEGER NOT NULL,
ID_ARTISTA_GERAL INTEGER NOT NULL,

CONSTRAINT TB_ARTISTA_PK PRIMARY KEY (ID_ARTISTA_GERAL,
ID_AGENTE)
);

--DROP TABLE TB_ARTISTA_FIXO;
CREATE TABLE TB_ARTISTA_FIXO(
    ID_ARTISTA_GERAL INTEGER NOT NULL,
    ID_ENDERECO INTEGER NOT NULL,
    TELEFONE VARCHAR(14),
    CPF VARCHAR(12),
    EMAIL VARCHAR(120),

CONSTRAINT TB_ARTISTA_FIXO_PK PRIMARY KEY (ID_ARTISTA_GERAL,
ID_ENDERECO)
);

--DROP TABLE TB_FORMA_PAGAMENTO;
CREATE TABLE TB_FORMA_PAGAMENTO(
    ID_FORMA_PAGAMENTO INTEGER NOT NULL,
    FORMA_PAGAMENTO VARCHAR(30) NOT NULL,

CONSTRAINT TB_FORMA_PAGAMENTO_PK PRIMARY KEY (ID_FORMA_PAGAMENTO)
);

--DROP TABLE TB_DATA_CONTA;
CREATE TABLE TB_DATA_CONTA(
    ID_DATA_CONTA INTEGER NOT NULL,
    DATA_CONTA DATETIME,
    TIPO_DATA VARCHAR(11) NOT NULL,

CONSTRAINT TB_DATA_CONTA_PK PRIMARY KEY (ID_DATA_CONTA)
);

--DROP TABLE TB_CONTAS;
CREATE TABLE TB_CONTAS(
    ID_CONTAS INTEGER NOT NULL,
    ID_DATA_CONTA INTEGER NOT NULL,
    TIPO_CONTA VARCHAR(15),
    DATA_LANCAMENTO DATETIME,
    DATA_ENTREGUE DATETIME,
    DESCRICAO VARCHAR(220),
    SITUACAO VARCHAR(30),
    DEPARTAMENTO VARCHAR(30),
    VALOR_TOTAL SMALLMONEY,
    ATIVO NUMERIC(1),

CONSTRAINT TB_CONTAS_PK PRIMARY KEY (ID_CONTAS),
CONSTRAINT TB_DATA_CONTA_FK FOREIGN KEY (ID_DATA_CONTA)
REFERENCES TB_DATA_CONTA (ID_DATA_CONTA)
);

--DROP TABLE TB_DETALHE_CONTA;
CREATE TABLE TB_DETALHE_CONTA (

```

```

ID_DETALHE INTEGER,
ID_CONTAS INTEGER,
ID_FORMA_PAGAMENTO INTEGER,
VALOR NUMERIC(24,2),

CONSTRAINT DETALHE_CONTA_PK PRIMARY KEY(ID_DETALHE),
CONSTRAINT DETALHE_CONTA_FK FOREIGN KEY(ID_CONTAS)
REFERENCES TB_CONTAS(ID_CONTAS),
CONSTRAINT DETALHE_FORMA_PAG_FK FOREIGN KEY(ID_FORMA_PAGAMENTO)
REFERENCES TB_FORMA_PAGAMENTO(ID_FORMA_PAGAMENTO)
);

--DROP TABLE TB_CLIENTE;
CREATE TABLE TB_CLIENTE(
    ID_CLIENTE INTEGER NOT NULL,
    CPF VARCHAR(12),
    ID_PESSOA INTEGER NOT NULL,
    EMAIL VARCHAR(120),

    CONSTRAINT TB_CLIENTE_PK PRIMARY KEY (ID_CLIENTE) ,
    CONSTRAINT TB_CLI_PESSOA_FK FOREIGN KEY(ID_PESSOA) REFERENCES
    TB_PESSOA(ID_PESSOA)
);

--DROP TABLE TB_TIPO_ASSENTO;
CREATE TABLE TB_TIPO_ASSENTO(
    ID_TIPO_ASSENTO NUMERIC(4) NOT NULL,
    TIPO_ASSENTO VARCHAR(8),

    CONSTRAINT TB_TIPO_ASSENTO_PK PRIMARY KEY (ID_TIPO_ASSENTO)
);

--DROP TABLE TB_SETOR;
CREATE TABLE TB_SETOR(
    ID_SETOR INTEGER NOT NULL,
    SETOR VARCHAR(6),

    CONSTRAINT TB_SETOR_PK PRIMARY KEY (ID_SETOR)
);

--DROP TABLE TB_ASSENTO;
CREATE TABLE TB_ASSENTO(
    ID_ASSENTO INTEGER NOT NULL,
    ASSENTO_NUMER VARCHAR(20) NOT NULL,
    ASSENTO_FILEIRA VARCHAR(1) NOT NULL,
    ID_TIPO_ASSENTO NUMERIC(4) NOT NULL,
    ID_SETOR INTEGER NOT NULL,

    CONSTRAINT TB_ASSENTO_PK PRIMARY KEY (ID_ASSENTO) ,
    CONSTRAINT TB_SETOR_FK FOREIGN KEY(ID_SETOR) REFERENCES
    TB_SETOR(ID_SETOR),
    CONSTRAINT TB_TIPO_ASSENTO_FK FOREIGN KEY(ID_TIPO_ASSENTO)
REFERENCES
    TB_TIPO_ASSENTO(ID_TIPO_ASSENTO)
);

```

```

--DROP TABLE TB_ASSENTO_CLI;
CREATE TABLE TB_ASSENTO_CLI(
    ID_ASSENTO INTEGER NOT NULL,
    ID_CLIENTE INTEGER NOT NULL,
    ID_EVENTO INTEGER NOT NULL,
    ATIVO NUMERIC NOT NULL,
    CONSTRAINT TB_ASSENTO_CLI_PK PRIMARY KEY
    (ID_ASSENTO, ID_CLIENTE, ID_EVENTO, ATIVO)
);

--DROP TABLE TB_EVENTO;
CREATE TABLE TB_EVENTO(
    ID_EVENTO INTEGER NOT NULL,
    TITULO VARCHAR(20),
    DATA_EVENTO DATE,
    DESCRICAO VARCHAR(141),
    HORARIO_INICIO DATETIME,
    HORARIO_FINAL DATETIME,
    DURACAO VARCHAR(5),
    N_ARTISTAS NUMERIC(3),
    TIPO_PAG VARCHAR(20),
    ATIVO NUMERIC(1) NOT NULL,
    VALOR_EVENTO NUMERIC(10,2),
    URL_IMG VARCHAR(250),
    CONSTRAINT TB_EVENTO_PK PRIMARY KEY (ID_EVENTO)
);

--DROP TABLE TB_EVENTO_INGRESSO;
CREATE TABLE TB_EVENTO_INGRESSO(
    ID_EI INTEGER NOT NULL,
    ID_EVENTO INTEGER,
    ID_PAGAMENTO INTEGER,
    PRECO_TOTAL SMALLMONEY,
    DESCONTO SMALLMONEY,
    PRECO_TOTAL_PAGO SMALLMONEY,
    TROCO SMALLMONEY,
    CONSTRAINT TB_EVENTOINGRESSO_PK PRIMARY KEY (ID_EI) ,
    CONSTRAINT TB_EVENTOINGRESSO_FK FOREIGN KEY (ID_EVENTO)
    REFERENCES TB_EVENTO (ID_EVENTO)
);

--DROP TABLE TB_VENDA_INGRESSO;
CREATE TABLE TB_VENDA_INGRESSO(
    ID_VENDA INTEGER NOT NULL,
    ID_PESSOA INTEGER,
    DATAVENDA DATETIME,
    SITUACAO VARCHAR(15),
    ATIVO NUMERIC(1),
    CONSTRAINT TB_VENDA_PESSOA_FK FOREIGN KEY (ID_PESSOA)
    REFERENCES TB_PESSOA (ID_PESSOA),

```

```

        CONSTRAINT TB_VENDA_INGRESSO_PK PRIMARY KEY (ID_VENDA)
) ;

--DROP TABLE TB_DETALHE_CONTA_VENDA;
CREATE TABLE TB_DETALHE_CONTA_VENDA(
ID_DETALHE_VENDA INTEGER,
ID_CONTAS INTEGER,
ID_VENDA INTEGER,

CONSTRAINT DETALHE_CONTA_VENDA_PK PRIMARY KEY(ID_DETALHE_VENDA),
CONSTRAINT DETALHE_CV_CONTA_FK FOREIGN KEY(ID_CONTAS)
REFERENCES TB_CONTAS(ID_CONTAS),
CONSTRAINT DETALHE_CV_VENDA_FK FOREIGN KEY(ID_VENDA)
REFERENCES TB_VENDA_INGRESSO(ID_VENDA)

) ;

--DROP TABLE TB_ITEM_VENDA;
CREATE TABLE TB_ITEM_VENDA(
    ID_ITEM_VENDA INTEGER NOT NULL,
    ID_VENDA INTEGER,
    ID_EI INTEGER,
    ID_ASSENTO INTEGER,

    CONSTRAINT TB_ITEM_VENDA_PK PRIMARY KEY (ID_ITEM_VENDA) ,
    CONSTRAINT TB_VENDA_FK FOREIGN KEY (ID_VENDA) REFERENCES
TB_VENDA_INGRESSO(ID_VENDA),
    CONSTRAINT TB_EI_FK FOREIGN KEY (ID_EI) REFERENCES
TB_EVENTO_INGRESSO(ID_EI)
) ;

--DROP TABLE TB_PAGAMENTO;
CREATE TABLE TB_PAGAMENTO(
    ID_PAGAMENTO_GERAL INTEGER NOT NULL,
    ID_PAGAMENTO INTEGER NOT NULL,
    ID_FORMA_PAGAMENTO INTEGER,
    QUANTIA NUMERIC(38,5),

    CONSTRAINT TB_PAGAMENTO_PK PRIMARY KEY
(ID_PAGAMENTO_GERAL, ID_PAGAMENTO),
    CONSTRAINT TB_PAGAMENTO_FK FOREIGN KEY(ID_FORMA_PAGAMENTO)
REFERENCES
    TB_FORMA_PAGAMENTO(ID_FORMA_PAGAMENTO)
) ;

--DROP TABLE TB_EVENTO_ARTISTA;
CREATE TABLE TB_EVENTO_ARTISTA(
    ID_EVENTO INTEGER NOT NULL,

```

```

    ID_PESSOA INTEGER NOT NULL,
    CONSTRAINT TB_EVENTO_PESSOA_PK PRIMARY KEY (ID_EVENTO, ID_PESSOA)
) ;

--DROP TABLE TB_PERGUNTASECRETA;
CREATE TABLE TB_PERGUNTASECRETA(
    ID_PERGUNTASECRETA INTEGER NOT NULL,
    DESCRICAO VARCHAR(128),
    RESPOSTA VARCHAR(120),
    CONSTRAINT TB_PERGUNTASECRETA_PK PRIMARY KEY (ID_PERGUNTASECRETA)
) ;

--DROP TABLE TB_NIVELACESSO;
CREATE TABLE TB_NIVELACESSO(
    ID_NIVELACESSO INTEGER NOT NULL,
    NIVELACESSO VARCHAR(20),
    CONSTRAINT TB_NIVELACESSO_PK PRIMARY KEY (ID_NIVELACESSO)
) ;

--DROP TABLE TB_LOGIN;
CREATE TABLE TB_LOGIN(
    ID_LOGIN INTEGER NOT NULL,
    ID_NIVELACESSO INTEGER NOT NULL,
    ID_PERGUNTASECRETA INTEGER NOT NULL,
    USUARIO VARCHAR(50) NOT NULL,
    SENHA VARCHAR(50) NOT NULL,
    DATAULTIMOACESSO DATE,
    ID_PESSOA INTEGER NOT NULL,
    CONSTRAINT TB_NIVEL_ACESSO_FK FOREIGN KEY (ID_NIVELACESSO)
        REFERENCES TB_NIVELACESSO(ID_NIVELACESSO),
    CONSTRAINT TB_PERGUNTASECRETA_FK FOREIGN KEY (ID_PERGUNTASECRETA)
        REFERENCES
            TB_PERGUNTASECRETA(ID_PERGUNTASECRETA),
    CONSTRAINT TB_LOGIN_PK PRIMARY KEY (ID_LOGIN)
) ;

--DROP TABLE TB_FUNCAO;
CREATE TABLE TB_FUNCAO(
    ID_FUNCAO INTEGER NOT NULL,
    FUNCAO VARCHAR(40) NOT NULL,
    CONSTRAINT TB_FUNCAO_PK PRIMARY KEY (ID_FUNCAO)
) ;

--DROP TABLE TB_FUNCIONARIO;
CREATE TABLE TB_FUNCIONARIO(
    ID_FUNC INTEGER NOT NULL,
    ID_FUNCAO INTEGER NOT NULL,
    ID_ENDERECO INTEGER,
    ID_PESSOA INTEGER NOT NULL,
    CPF VARCHAR(12),

```

```

EMAIL VARCHAR(120),
TELEFONE VARCHAR(14),

CONSTRAINT TB_FUNC_PESSOA_FK FOREIGN KEY (ID_PESSOA)
REFERENCES TB_PESSOA(ID_PESSOA),
CONSTRAINT TB_FUNC_ENDERECHO_FK FOREIGN KEY (ID_ENDERECHO)
REFERENCES TB_ENDERECHO(ID_ENDERECHO),
CONSTRAINT TB_FUNCAO_FK FOREIGN KEY (ID_FUNCAO)
REFERENCES TB_FUNCAO(ID_FUNCAO),
CONSTRAINT TB_FUNCIONARIO_PK PRIMARY KEY (ID_FUNC)
);

--DROP TABLE TB_CANCELAVENDA;
CREATE TABLE TB_CANCELAVENDA(
ID_CANCELAVENDA INTEGER NOT NULL,
ID_VENDA INTEGER NOT NULL,
MOTIVO VARCHAR(20),
DATA_CANCELAMENTO DATE,
ID_FUNC INTEGER NOT NULL,

CONSTRAINT TB_CANCELAVENDA_PK PRIMARY KEY (ID_CANCELAVENDA),
CONSTRAINT TB_CANCELAVENDA_FK FOREIGN KEY (ID_VENDA)
    REFERENCES TB_VENDA_INGRESSO(ID_VENDA),
CONSTRAINT TB_CANCELAVENDA_FUNC_FK FOREIGN KEY (ID_FUNC)
    REFERENCES TB_FUNCIONARIO(ID_FUNC)

);

--DROP TABLE TB_PRECOS;
CREATE TABLE TB_PRECOS(
ID_SETOR INTEGER NOT NULL,
ID_TIPO_ASSENTO INTEGER NOT NULL,
PERC NUMERIC(3),

CONSTRAINT TB_PRECOS_PK PRIMARY KEY (ID_TIPO_ASSENTO, ID_SETOR)
);

--DROP TABLE TB_BANNER;
CREATE TABLE TB_BANNER(
ID_BANNER INTEGER NOT NULL,
URL_IMG VARCHAR(250),
INDEX_BANNER NUMERIC(2),
ATIVO NUMERIC(1),

CONSTRAINT TB_BANNER_PK PRIMARY KEY (ID_BANNER)
);

--DROP TABLE TB_DESCONTOS;
CREATE TABLE TB_DESCONTOS(
ID_DESCONTO INTEGER NOT NULL,
DESCONTO NUMERIC(5,1),
COD_PROMOCIONAL VARCHAR(20),
ATIVO NUMERIC(1),
N_USOS NUMERIC(5),

```

```

        CONSTRAINT TB_DESCONTOS_PK PRIMARY KEY(ID_DESCONTO)
) ;

--DROP TABLE TB_DESCONTOS_CLIENTE;
CREATE TABLE TB_DESCONTOS_CLIENTE(
    ID_DESCONTO INTEGER NOT NULL,
    ID_CLIENTE INTEGER NOT NULL,

        CONSTRAINT TB_DESCONTOS_CLIENTE_PK PRIMARY KEY(ID_DESCONTO,
ID_CLIENTE)
) ;

--DROP TABLE TB_TIPO_LOG;
CREATE TABLE TB_TIPO_LOG(
    ID_TIPO_LOG INTEGER NOT NULL,
    TIPO_LOG VARCHAR(15),

CONSTRAINT TB_TIPO_LOG_PK PRIMARY KEY(ID_TIPO_LOG)
) ;

--DROP TABLE TB_LOG;
CREATE TABLE TB_LOG(
    ID_LOG INTEGER NOT NULL,
    DATA_LOG DATETIME,
    ID_TIPO_LOG INTEGER NOT NULL,
    ID_FUNC INTEGER NOT NULL,

CONSTRAINT TB_LOG_PK PRIMARY KEY(ID_LOG),
CONSTRAINT TB_TIPO_LOG_FK FOREIGN KEY (ID_TIPO_LOG)
    REFERENCES TB_TIPO_LOG(ID_TIPO_LOG),
CONSTRAINT TB_FUNC_LOG_FK FOREIGN KEY (ID_FUNC)
    REFERENCES TB_FUNCIONARIO(ID_FUNC)
) ;

--DROP TABLE TB_TABELAS_LOG;
CREATE TABLE TB_TABELAS_LOG(
    ID_TABELA_LOG INTEGER NOT NULL,
    TABELA VARCHAR(15),
    TABELA_APELIDO VARCHAR(15),

CONSTRAINT TB_TABELA_LOG PRIMARY KEY(ID_TABELA_LOG)
) ;

--DROP TABLE TB_LOG_INSERCAO
CREATE TABLE TB_LOG_INSERCAO(
    ID_LOG_INSER INTEGER NOT NULL,
    ID_LOG INTEGER NOT NULL,
    ID_TABELA_LOG INTEGER NOT NULL,
    ID_INSERIDO INTEGER NOT NULL,

CONSTRAINT TB_LOG_INSERCAO_PK PRIMARY KEY(ID_LOG_INSER),
CONSTRAINT TB_LOG_INSER_FK FOREIGN KEY (ID_LOG)
    REFERENCES TB_LOG(ID_LOG),
CONSTRAINT TABELA_LOG_INSER_FK FOREIGN KEY (ID_TABELA_LOG)

```

```

    REFERENCES TB_TABELAS_LOG(ID_TABELA_LOG)
) ;

ALTER TABLE TB_LOG_EXCLUSAO ADD DESCRICAO VARCHAR(50);
--DROP TABLE TB_LOG_EXCLUSAO
CREATE TABLE TB_LOG_EXCLUSAO(
ID_LOG_EXCL INTEGER NOT NULL,
ID_LOG INTEGER NOT NULL,
ID_TABELA_LOG INTEGER NOT NULL,
ID_EXCLUIDO INTEGER NOT NULL,
DESCRICAO VARCHAR(50),

CONSTRAINT TB_LOG_EXCLUSAO_PK PRIMARY KEY(ID_LOG_EXCL),
CONSTRAINT TB_LOG_EXCL_FK FOREIGN KEY (ID_LOG)
    REFERENCES TB_LOG(ID_LOG),
CONSTRAINT TABELA_LOG_EXCL_FK FOREIGN KEY (ID_TABELA_LOG)
    REFERENCES TB_TABELAS_LOG(ID_TABELA_LOG)
);

--DROP TABLE TB_LOG_ALTERACAO;
CREATE TABLE TB_LOG_ALTERACAO(
ID_LOG_ALTER INTEGER NOT NULL,
ID_LOG INTEGER NOT NULL,
ID_TABELA_LOG INTEGER NOT NULL,
ID_ALTERADO INTEGER NOT NULL,

CONSTRAINT TB_LOG_ALTERACAO_PK PRIMARY KEY(ID_LOG_ALTER),
CONSTRAINT TB_LOG_ALTER_FK FOREIGN KEY (ID_LOG)
    REFERENCES TB_LOG(ID_LOG),
CONSTRAINT TABELA_LOG_ALTER_FK FOREIGN KEY (ID_TABELA_LOG)
    REFERENCES TB_TABELAS_LOG(ID_TABELA_LOG)
);

CREATE TABLE TB_BANDEIRA(
ID_BANDEIRA INTEGER NOT NULL,
DESC_BANDEIRA VARCHAR(20) NOT NULL,

CONSTRAINT TB_BANDEIRA_PK PRIMARY KEY(ID_BANDEIRA)
)

CREATE TABLE TB_DETALHE_PGT_CRED(
ID_PGT_CRED INTEGER NOT NULL,
ID_PAGAMENTO INTEGER NOT NULL,
ID_PAGAMENTO_GERAL INTEGER NOT NULL,
ID_BANDEIRA INTEGER NOT NULL,
QTDE_PARCELAS NUMERIC(2) NOT NULL,

CONSTRAINT TB_PGT_CRED PRIMARY KEY(ID_PGT_CRED),
CONSTRAINT TB_PG_DTCRED_PGT_FK FOREIGN KEY
    (ID_PAGAMENTO_GERAL, ID_PAGAMENTO)
REFERENCES TB_PAGAMENTO(ID_PAGAMENTO_GERAL, ID_PAGAMENTO),
CONSTRAINT TB_BD_DTCRED_PGT_FK FOREIGN KEY (ID_BANDEIRA)
REFERENCES TB_BANDEIRA(ID_BANDEIRA)
)

CREATE TABLE TB_DETALHE_PGT_DEB(
ID_PGT_DEB INTEGER NOT NULL,

```

```

ID_PAGAMENTO INTEGER NOT NULL,
ID_PAGAMENTO_GERAL INTEGER NOT NULL,
ID_BANDEIRA INTEGER NOT NULL,

CONSTRAINT TB_PGT_DEB PRIMARY KEY(ID_PGTO_DEB),
CONSTRAINT TB_PG_DTDEB_PGT_FK FOREIGN KEY
(ID_PAGAMENTO_GERAL, ID_PAGAMENTO)
REFERENCES TB_PAGAMENTO(ID_PAGAMENTO_GERAL, ID_PAGAMENTO),
CONSTRAINT TB_BD_DTDEB_PGT_FK FOREIGN KEY (ID_BANDEIRA)
REFERENCES TB_BANDEIRA(ID_BANDEIRA)
)

DROP TABLE TB_FEEDBACK
CREATE TABLE TB_FEEDBACK(
ID_FEEDBACK INTEGER NOT NULL,
MENSAGEM VARCHAR(120),
TITULO VARCHAR(60),
AVALIACAO NUMERIC(1),
ID_PESSOA INTEGER NOT NULL,
DATA_ENVIO DATETIME,

CONSTRAINT TB_FEEDBACK_PK PRIMARY KEY (ID_FEEDBACK),
CONSTRAINT TB_FEEDBACK_FK FOREIGN KEY(ID_PESSOA)
REFERENCES TB_PESSOA(ID_PESSOA)
)

```

PROCEDURES

```

/*
    COMEDY HOUSE

    Tema: CASA DE SHOW DE STAND-UP

    Turma: INF3AM

    Grupo: Arthur Porto          01
           Diego Gomes            07
           Gabriel Leandro         11
           Lucas Camargo           19
           Marcos Santos           26
*/

-----
--PESSOA, LOGIN, ENDERECO, FORMA DE PAGAMENTO, CLIENTE*, ARTISTA*,  

CONTAS*, AGENTE, FUNCAO, FUNCIONARIO*, EVENTO*
-----

-
create or alter procedure SP_PESSOA
(
@ID_PESSOA integer,
@NOME varchar(150),
@DATA_NASC date,
@SEXO varchar(9),
@ATIVO numeric(1,0),

```

```

@ID_TIPO_PESSOA integer,
@OPR char(1)
AS
BEGIN
    BEGIN TRY
        IF(@OPR='D') -- DELETAR
        BEGIN
            DELETE FROM TB_PESSOA WHERE ID_PESSOA =
@ID_PESSOA;
            IF(@@ROWCOUNT= 0)
                RAISERROR('Erro inesperado, favor
contatar o gerente!', 18, 1)

        END
        ELSE
            IF(@OPR='A') -- ALTERAR
            BEGIN
                UPDATE TB_PESSOA SET NOME = @NOME, DATA_NASC =
@DATA_NASC, SEXO = @SEXO , ATIVO = @ATIVO, ID_TIPO_PESSOA =
@ID_TIPO_PESSOA
                WHERE ID_PESSOA = @ID_PESSOA;
            END
            ELSE
                IF(@OPR='I') -- INCLUIR
                BEGIN
                    INSERT INTO TB_PESSOA(NOME, DATA_NASC, SEXO,
ATIVO, ID_TIPO_PESSOA, DATA_CRIACAO)
                    VALUES (@NOME, @DATA_NASC, @SEXO,
@ATIVO,@ID_TIPO_PESSOA, GETDATE());
                END
        END TRY
        BEGIN CATCH
            IF(@@ROWCOUNT=0)
                RAISERROR('Erro! Nenhuma linha para ser modificado!', 17, 1)
            ELSE

                RAISERROR('Erro inesperado, favor contatar o gerente!', 18, 1)
                return
        END CATCH
    END;

    SELECT * FROM TB_PESSOA WHERE ID_PESSOA=69;

    SP_HELPTEXT SP_PESSOA;
    EXEC SP_PESSOA 100, 'GIL' , '06/08/1984' , 'MASCULINO', 0,2, 'D';
-----
create or alter procedure SP_CLIENTE
(
@ID_CLIENTE integer,
@ID_PESSOA integer,
@CPF varchar(12),
@email varchar(120),
@OPR char(1)
AS
DECLARE @PESSOA integer;
BEGIN

```

```

BEGIN TRY
    IF(@OPR='D') -- DELETAR
        BEGIN
            DELETE FROM TB_PERGUNTASECRETA WHERE
ID_PERGUNTASECRETA = (SELECT ID_PERGUNTASECRETA FROM
TB_PERGUNTASECRETA WHERE
ID_PERGUNTASECRETA = (SELECT ID_PERGUNTASECRETA
FROM TB_LOGIN WHERE ID_LOGIN= (SELECT ID_LOGIN FROM TB_LOGIN WHERE
ID_PESSOA=(SELECT ID_PESSOA FROM TB_CLIENTE WHERE
ID_CLIENTE = @ID_CLIENTE))));

            DELETE FROM TB_LOGIN WHERE ID_PESSOA = (SELECT
ID_PESSOA FROM TB_CLIENTE WHERE
ID_CLIENTE = @ID_CLIENTE);

            DELETE FROM TB_DESCONTOS_CLIENTE WHERE ID_CLIENTE =
@ID_CLIENTE;
            DELETE FROM TB_ASSENTO_CLI WHERE ID_CLIENTE=
@ID_CLIENTE;

            SET @PESSOA= ( SELECT ID_PESSOA FROM TB_CLIENTE WHERE
ID_CLIENTE = @ID_CLIENTE);
            DELETE FROM TB_CLIENTE WHERE ID_CLIENTE =
@ID_CLIENTE;

            DELETE FROM TB_PESSOA WHERE ID_PESSOA = @PESSOA;

        END
    ELSE
        IF(@OPR='A') -- ALTERAR
            BEGIN
                UPDATE TB_CLIENTE SET CPF = @CPF, EMAIL =
@EMAIL WHERE ID_CLIENTE = @ID_CLIENTE;
            END
        ELSE
            IF(@OPR='I') -- INCLUIR
                BEGIN
                    INSERT INTO TB_CLIENTE(CPF, ID_PESSOA, EMAIL)
VALUES (@CPF, (SELECT MAX(ID_PESSOA) FROM
TB_PESSOA), @EMAIL);
                END
        END TRY
        BEGIN CATCH
            RAISERROR('Erro inesperado, favor contatar o
gerente!',18, 1)
            return
        END CATCH
    END;
-----  

CREATE OR ALTER procedure SP_ARTISTA
(
@ID_ARTISTA_GERAL integer,

```

```

@ID_REDESOCIAL integer,
@FACEBOOK varchar(120),
@TWITTER varchar(120),
@INSTAGRAM varchar(120),
@ID_PESSOA integer,
@ID_TIPO_PESSOA integer,
@URL_IMG varchar(250),
@ID_AGENTE integer,
@ID_ENDERECHO integer,
@TELEFONE varchar(14),
@CPF varchar(12),
@EMAIL varchar(120),
@OPR char(1))
AS
BEGIN
    BEGIN TRY
        IF(@ID_TIPO_PESSOA=4) --ARTISTA FIXO
            BEGIN
                IF(@OPR='D') -- DELETAR
                    BEGIN
                        DELETE FROM TB_ARTISTA WHERE
ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL;

                        DELETE FROM TB_ARTISTA_FIXO WHERE
ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL;

                        SET @ID_PESSOA=(SELECT ID_PESSOA FROM
TB_ARTISTA_GERAL WHERE
ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL);

                        SET @ID_REDESOCIAL= (SELECT ID_REDESOCIAL FROM
TB_ARTISTA_GERAL WHERE
ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL);

                        SET @ID_ENDERECHO= (SELECT ID_ENDERECHO FROM
TB_ARTISTA_FIXO WHERE
ID_ARTISTA_GERAL =
@ID_ARTISTA_GERAL);

                        DELETE FROM TB_ARTISTA_GERAL WHERE
ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL;

                        DELETE FROM TB_PESSOA WHERE ID_PESSOA
=@ID_PESSOA;

                        DELETE FROM TB_REDESOCIAL WHERE ID_REDESOCIAL=
@ID_REDESOCIAL;

                        DELETE FROM TB_ENDERECHO WHERE
ID_ENDERECHO= @ID_ENDERECHO;
                    END
                ELSE
                    IF(@OPR='A') -- ALTERAR

```

```

        BEGIN
UPDATE TB_ARTISTA_GERAL SET ID_REDESOCIAL =
@ID_REDESOCIAL,
URL_IMG = @URL_IMG
WHERE ID_PESSOA = @ID_PESSOA;

UPDATE TB_REDESOCIAL SET FACEBOOK = @FACEBOOK,
TWITTER = @TWITTER,
INSTAGRAM = @INSTAGRAM WHERE ID_REDESOCIAL =
@ID_REDESOCIAL;

UPDATE TB_ARTISTA_FIXO SET ID_ENDERECO =
@ID_ENDERECO, TELEFONE = @TELEFONE,
CPF = @CPF, EMAIL = @EMAIL
WHERE ID_ARTISTA_GERAL = @ID_ARTISTA_GERAL;
END
ELSE
IF (@OPR='AN') -- ALTERAR DE ARTISTA PRA
ARTISTA FIXO
BEGIN
DELETE FROM TB_ARTISTA WHERE ID_ARTISTA_GERAL =
@ID_ARTISTA_GERAL;

UPDATE TB_ARTISTA_GERAL SET ID_REDESOCIAL =
@ID_REDESOCIAL,
URL_IMG = @URL_IMG
WHERE ID_PESSOA = @ID_PESSOA;

UPDATE TB_REDESOCIAL SET FACEBOOK= @FACEBOOK,
TWITTER=@TWITTER,
INSTAGRAM= @INSTAGRAM WHERE ID_REDESOCIAL=
@ID_REDESOCIAL;

INSERT INTO TB_ARTISTA_FIXO(ID_ARTISTA_GERAL,
ID_ENDERECO, TELEFONE, CPF, EMAIL)
VALUES (@ID_ARTISTA_GERAL, (SELECT
MAX(ID_ENDERECO) FROM TB_ENDERECO), @TELEFONE,
@CPF, @EMAIL);
END
ELSE
IF (@OPR='I') -- INCLUIR
BEGIN
INSERT INTO TB_REDESOCIAL(FACEBOOK, TWITTER,
INSTAGRAM)
VALUES (@FACEBOOK, @TWITTER, @INSTAGRAM);

INSERT INTO TB_ARTISTA_GERAL(ID_REDESOCIAL,
ID_PESSOA, URL_IMG)
VALUES ((SELECT MAX(ID_REDESOCIAL) FROM
TB_REDESOCIAL) , (SELECT MAX(ID_PESSOA) FROM TB_PESSOA) ,@URL_IMG);

INSERT INTO TB_ARTISTA_FIXO(ID_ARTISTA_GERAL,
ID_ENDERECO, TELEFONE, CPF, EMAIL)
VALUES ((SELECT MAX(ID_ARTISTA_GERAL) FROM
TB_ARTISTA_GERAL) , (SELECT MAX(ID_ENDERECO) FROM TB_ENDERECO),
@TELEFONE,

```

```

@CPF, @EMAIL);
END
END

ELSE -- ARTISTA
IF (@OPR='D') -- DELETAR
BEGIN
DELETE FROM TB_ARTISTA WHERE ID_ARTISTA_GERAL=
@ID_ARTISTA_GERAL;

DELETE FROM TB_ARTISTA_FIXO WHERE
ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL;

SET @ID_PESSOA=(SELECT ID_PESSOA FROM
TB_ARTISTA_GERAL WHERE
ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL);

SET @ID_REDESOCIAL= (SELECT ID_REDESOCIAL FROM
TB_ARTISTA_GERAL WHERE
ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL);

DELETE FROM TB_ARTISTA_GERAL WHERE
ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL;

DELETE FROM TB_PESSOA WHERE ID_PESSOA
=@ID_PESSOA;

DELETE FROM TB_REDESOCIAL WHERE ID_REDESOCIAL=
@ID_REDESOCIAL;

END
ELSE
IF (@OPR='A') -- ALTERAR
BEGIN
UPDATE TB_ARTISTA_GERAL SET ID_REDESOCIAL =
@ID_REDESOCIAL, URL_IMG = @URL_IMG
WHERE ID_PESSOA = @ID_PESSOA;

UPDATE TB_REDESOCIAL SET FACEBOOK = @FACEBOOK,
TWITTER = @TWITTER,
INSTAGRAM= @INSTAGRAM WHERE ID_REDESOCIAL =
@ID_REDESOCIAL;

UPDATE TB_ARTISTA SET ID_AGENTE= @ID_AGENTE
WHERE ID_ARTISTA_GERAL = @ID_ARTISTA_GERAL;
END
ELSE
IF (@OPR='AN') -- ALTERAR DE ARTISTA FIXO PARA ARTISTA
BEGIN
DELETE FROM TB_ENDERECHO WHERE ID_ENDERECHO=
(SELECT ID_ENDERECHO FROM TB_ARTISTA_FIXO

```

```

        WHERE ID_ARTISTA_GERAL= @ID_ARTISTA_GERAL);

        DELETE FROM TB_ARTISTA_FIXO WHERE ID_ARTISTA_GERAL=
@ID_ARTISTA_GERAL;

        UPDATE TB_ARTISTA_GERAL SET ID_REDESOCIAL=
@ID_REDESOCIAL, URL_IMG= @URL_IMG
        WHERE ID_PESSOA = @ID_PESSOA;

        UPDATE TB_REDESOCIAL SET FACEBOOK = @FACEBOOK,
TWITTER = @TWITTER,
                INSTAGRAM = @INSTAGRAM WHERE ID_REDESOCIAL =
@ID_REDESOCIAL;

        INSERT INTO TB_ARTISTA(ID_ARTISTA_GERAL, ID_AGENTE)
VALUES (@ID_ARTISTA_GERAL,@ID_AGENTE);
        END
        ELSE
        IF(@OPR='I') -- INCLUIR
        BEGIN
                INSERT INTO TB_REDESOCIAL(FACEBOOK,TWITTER,
INSTAGRAM)
                VALUES (@FACEBOOK, @TWITTER, @INSTAGRAM);

                INSERT INTO TB_ARTISTA_GERAL(ID_REDESOCIAL,
ID_PESSOA, URL_IMG)
                VALUES ((SELECT MAX(ID_REDESOCIAL) FROM
TB_REDESOCIAL), (SELECT MAX(ID_PESSOA) FROM TB_PESSOA), @URL_IMG);

                INSERT INTO TB_ARTISTA(ID_ARTISTA_GERAL,
ID_AGENTE)
                VALUES ((SELECT MAX(ID_ARTISTA_GERAL) FROM
TB_ARTISTA_GERAL),@ID_AGENTE);
                END
        END TRY
        BEGIN CATCH
                RAISERROR('Erro inesperado, favor contatar o
gerente!',18, 1)
                return
        END CATCH
END;
-----
CREATE OR ALTER procedure SP_AGENTE
(
@ID_AGENTE integer,
@ID_ENDERECHO integer,
@TELEFONE varchar(14),
@TIPO_PESSOA varchar(8),
@DOCUMENTO varchar(18),
@EMAIL varchar(120),
@NOME_PRINCIPAL varchar(120),
@NOME_SECUNDARIO varchar(120),
@ATIVO NUMERIC(1,0),
@OPR char(1))
AS

```

```

BEGIN
    BEGIN TRY
        IF (@OPR='D') -- DELETAR
            BEGIN
                DELETE FROM TB_ENDERECHO WHERE ID_ENDERECHO= (SELECT
ID_ENDERECHO FROM TB_AGENTE
                    WHERE ID_AGENTE= @ID_AGENTE);
                DELETE FROM TB_AGENTE WHERE ID_AGENTE = @ID_AGENTE;
                IF (@@ROWCOUNT= 0)
                    RAISERROR('Erro inesperado, favor
contatar o gerente!', 18, 1)
            END
        ELSE
            IF (@OPR='A') -- ALTERAR
                BEGIN
                    UPDATE TB_AGENTE SET TELEFONE= @TELEFONE,
TIPO_PESSOA= @TIPO_PESSOA, DOCUMENTO= @DOCUMENTO, EMAIL= @EMAIL,
NOME_PRINCIPAL= @NOME_PRINCIPAL, NOME_SECUNDARIO=
@NOME_SECUNDARIO, ATIVO = @ATIVO
                    WHERE ID_AGENTE = @ID_AGENTE;
                END
            ELSE
                IF (@OPR='I') -- INCLUIR
                    BEGIN
                        INSERT INTO TB_AGENTE (ID_ENDERECHO, TELEFONE,
TIPO_PESSOA, DOCUMENTO, EMAIL, NOME_PRINCIPAL, NOME_SECUNDARIO,
ATIVO, DATA_CRIACAO)
                        VALUES ((SELECT MAX(ID_ENDERECHO) FROM TB_ENDERECHO),
@TELEFONE, @TIPO_PESSOA,
@DOCUMENTO, @EMAIL,
@NOME_PRINCIPAL, @NOME_SECUNDARIO, @ATIVO, GETDATE() );
                    END
                END TRY
                BEGIN CATCH
                    IF (@@ROWCOUNT=0)
                        RAISERROR('Erro! Nenhuma linha para ser modificado!'
, 17, 1)
                    ELSE
                        RAISERROR('Erro inesperado, favor contatar o
gerente!', 18, 1)
                    return
                END CATCH
            END;
        EXEC SP_AGENTE 1, 1, 40028922, 'FÍSICA', '09967523824',
'LUIZLSOUZA@GMAIL.COM', 'LUIZINHO', 'LUIZ', 1, 'A';
-----
CREATE OR ALTER procedure SP_ENDERECHO
(
@ID_ENDERECHO integer,
@CEP varchar(8),
@NUMERO numeric(6,0),
@COMPLEMENTO varchar(120),
@OPR char(1))
AS
BEGIN
    BEGIN TRY

```

```

    IF (@OPR='D') -- DELETAR
        BEGIN
            DELETE FROM TB_ENDERECHO WHERE ID_ENDERECHO =
@ID_ENDERECHO;
            IF (@@ROWCOUNT= 0)
                RAISERROR('Erro inesperado, favor contatar o
gerente!',18, 1)
        END
    ELSE
        IF (@OPR='A') -- ALTERAR
            BEGIN
                UPDATE TB_ENDERECHO SET CEP = @CEP, NUMERO = @NUMERO,
COMPLEMENTO = @COMPLEMENTO
                WHERE ID_ENDERECHO = @ID_ENDERECHO;
            END
        ELSE
            IF (@OPR='I') -- INCLUIR
                BEGIN
                    INSERT INTO TB_ENDERECHO(CEP,NUMERO,COMPLEMENTO)
VALUES (@CEP, @NUMERO , @COMPLEMENTO);
                END
            END TRY
            BEGIN CATCH
                IF (@@ROWCOUNT=0)
                    RAISERROR('Erro! Nenhuma linha para ser modificado!'
,17, 1)
                ELSE
                    RAISERROR('Erro inesperado, favor contatar o
gerente!',18, 1)
                RETURN
            END CATCH
        END;
    EXEC SP_ENDERECHO 1, '06413040', '210', NULL, 'A';
-----
create or ALTER procedure SP_LOGIN
(
@ID_LOGIN integer,
@ID_NIVELACESSO integer,
@ID_PERGUNTASECRETA integer,
@USUARIO varchar(50),
@SENHA varchar(50),
@DATAULTIMOACESSO date,
@ID_PESSOA integer,
@DESCRICAO varchar(128),
@RESPOSTA varchar(120),
@OPR char(1))
AS
BEGIN
    BEGIN TRY
        IF (@OPR='D') -- DELETAR
            BEGIN
                DELETE FROM TB_LOGIN WHERE ID_LOGIN =
@ID_LOGIN;

                DELETE FROM TB_PERGUNTASECRETA WHERE
ID_PERGUNTASECRETA = @ID_PERGUNTASECRETA;
            END
    END TRY
    BEGIN CATCH
        IF (@@ROWCOUNT=0)
            RAISERROR('Erro! Nenhuma linha para ser modificado!'
,17, 1)
        ELSE
            RAISERROR('Erro inesperado, favor contatar o
gerente!',18, 1)
        RETURN
    END CATCH
END;

```

```

        END
    ELSE
        IF(@OPR='A') -- ALTERAR
            BEGIN
                UPDATE TB_LOGIN SET ID_NIVELACESSO =
@ID_NIVELACESSO, USUARIO = @USUARIO, SENHA = @SENHA
                    WHERE ID_LOGIN = @ID_LOGIN;

                UPDATE TB_PERGUNTASECRETA SET DESCRICAO=
@DESCRICAO, RESPOSTA= @RESPOSTA
                    WHERE ID_PERGUNTASECRETA = (SELECT
ID_PERGUNTASECRETA FROM TB_LOGIN WHERE ID_LOGIN = @ID_LOGIN);
            END
        ELSE
            IF(@OPR='N') -- ALTERAR
                BEGIN
                    UPDATE TB_LOGIN SET ID_NIVELACESSO =
@ID_NIVELACESSO, USUARIO= @USUARIO, SENHA= @SENHA
                        WHERE ID_LOGIN = @ID_LOGIN;
                END
        ELSE
            IF(@OPR='I') -- INCLUIR
                BEGIN
                    INSERT INTO TB_PERGUNTASECRETA(DESCRICAO,
RESPOSTA)
                        VALUES (@DESCRICAO,@RESPOSTA);

                    INSERT INTO
TB_LOGIN(ID_NIVELACESSO, ID_PERGUNTASECRETA, USUARIO, SENHA,
DATAULTIMOACESSO, ID_PESSOA)
                        VALUES (@ID_NIVELACESSO, (SELECT
MAX(ID_PERGUNTASECRETA) FROM TB_PERGUNTASECRETA),
@USUARIO, @SENHA, NULL, (SELECT MAX(ID_PESSOA)
FROM TB_PESSOA) );
                END
            END TRY
            BEGIN CATCH
                IF(@@ROWCOUNT=0)
                    RAISERROR('Erro! Nenhuma linha para ser modificado!',17, 1)
                ELSE
                    RAISERROR('Erro inesperado, favor contatar o gerente!',18, 1)
                RETURN
            END CATCH
        END;
        EXEC SP_LOGIN 2, 3, 1, 'GIL', '123', '2018-08-17', 54, '1+2+3',
'123', 'A';
-----
CREATE OR ALTER procedure SP_CONTAS
(
@ID_CONTAS integer,
@ID_DATA_CONTA integer,
@DATA_CONTA date,
@TIPO_DATA varchar(),

```

```

@TIPO_CONTA varchar(15),
@DATA_LANCAMENTO date,
@DATA_ENTREGUE date,
@DESCRICAO varchar(220),
@SITUACAO varchar(30),
@DEPARTAMENTO varchar(30),
@VALOR_TOTAL numeric,
@ATIVO numeric(1,0),
@OPR char(1)
AS
BEGIN
    BEGIN TRY
        IF (@OPR='D') -- DELETAR
            BEGIN
                DELETE FROM TB_DETALHE_CONTA WHERE ID_CONTAS =
@ID_CONTAS;

                DELETE FROM TB_DETALHE_CONTA_VENDA WHERE ID_CONTAS =
@ID_CONTAS;

                DELETE FROM TB_CONTAS WHERE ID_CONTAS = @ID_CONTAS;

                DELETE FROM TB_DATA_CONTA WHERE ID_DATA_CONTA =
@ID_DATA_CONTA;
                IF (@@ROWCOUNT= 0)
                    RAISERROR('Erro inesperado, favor
contatar o gerente!', 18, 1)
                END
            ELSE
                IF (@OPR='A') -- ALTERAR
                    BEGIN
                        UPDATE TB_CONTAS SET TIPO_CONTA = @TIPO_CONTA,
DESCRICAO= @DESCRICAO,
SITUACAO = @SITUACAO, DATA_ENTREGUE =
@DATA_ENTREGUE, DEPARTAMENTO= @DEPARTAMENTO, VALOR_TOTAL=
@VALOR_TOTAL, ATIVO = @ATIVO
                        WHERE ID_CONTAS = @ID_CONTAS;

                        UPDATE TB_DATA_CONTA SET DATA_CONTA = @DATA_CONTA,
TIPO_DATA = @TIPO_DATA
                        WHERE ID_DATA_CONTA = @ID_DATA_CONTA;
                    END
                ELSE
                    IF (@OPR='I') -- INCLUIR
                        BEGIN
                            INSERT INTO TB_DATA_CONTA(DATA_CONTA, TIPO_DATA)
VALUES (@DATA_CONTA,@TIPO_DATA);

                            INSERT INTO TB_CONTAS(ID_DATA_CONTA, TIPO_CONTA,
DATA_LANCAMENTO, DATA_ENTREGUE, DESCRICAO, SITUACAO,
DEPARTAMENTO, VALOR_TOTAL, ATIVO)
                            VALUES ((SELECT MAX(ID_DATA_CONTA) FROM
TB_DATA_CONTA),@TIPO_CONTA,
GETDATE(), @DATA_ENTREGUE , @DESCRICAO,
@SITUACAO, @DEPARTAMENTO, @VALOR_TOTAL, @ATIVO );
                        END
                    END
                END
            END
        END TRY
        BEGIN CATCH
            IF @@TRANCOUNT > 0
                ROLLBACK TRANSACTION
            RAISERROR('Erro inesperado, favor
contatar o gerente!', 18, 1)
        END CATCH
    END
END

```

```

END TRY
BEGIN CATCH
    IF (@@ROWCOUNT=0)
        RAISERROR('Erro! Nenhuma linha para ser modificada!', 17, 1)
    ELSE
        RAISERROR('Erro inesperado, favor contatar o gerente!', 18, 1)
        return
END CATCH
END;

-----
CREATE OR ALTER procedure SP_FORMA_PAGAMENTO
(
@ID_FORMA_PAGAMENTO integer,
@FORMA_PAGAMENTO varchar(30),
@OPR char(1))
AS
BEGIN
    BEGIN TRY
        IF (@OPR='D') -- DELETAR
            BEGIN
                DELETE FROM TB_PAGAMENTO WHERE ID_PAGAMENTO IN
                (SELECT ID_PAGAMENTO FROM TB_PAGAMENTO WHERE ID_FORMA_PAGAMENTO=
                @ID_FORMA_PAGAMENTO);
                DELETE FROM TB_DETALHE_CONTA WHERE ID_DETALHE IN
                (SELECT ID_DETALHE FROM TB_DETALHE_CONTA WHERE ID_FORMA_PAGAMENTO=
                @ID_FORMA_PAGAMENTO);
                DELETE FROM TB_FORMA_PAGAMENTO WHERE
                ID_FORMA_PAGAMENTO = @ID_FORMA_PAGAMENTO;
                IF (@@ROWCOUNT=0)
                    RAISERROR('Erro inesperado, favor contatar o gerente!', 18, 1)
            END
        ELSE
            IF (@OPR='A') -- ALTERAR
                BEGIN
                    UPDATE TB_FORMA_PAGAMENTO SET FORMA_PAGAMENTO =
                    @FORMA_PAGAMENTO
                    WHERE ID_FORMA_PAGAMENTO = @ID_FORMA_PAGAMENTO;
                END
            ELSE
                IF (@OPR='I') -- INCLUIR
                    BEGIN
                        INSERT INTO TB_FORMA_PAGAMENTO(FORMA_PAGAMENTO)
                        VALUES (@FORMA_PAGAMENTO);
                    END
    END TRY
    BEGIN CATCH
        IF (@@ROWCOUNT=0)
            RAISERROR('Erro! Nenhuma linha para ser modificada!', 17, 1)
        ELSE
            RAISERROR('Erro inesperado, favor contatar o gerente!', 18, 1)
    END CATCH
END;

```

```

        RETURN
    END CATCH
END;
EXEC SP_FORMA_PAGAMENTO 6, 'PAYPAL', 'I';
-----
CREATE OR ALTER procedure SP_FUNCAO
(
@ID_FUNCAO integer,
@FUNCAO varchar(40),
@OPR char(1))
AS
BEGIN
    BEGIN TRY
        IF (@OPR='D') -- DELETAR
            BEGIN
                DELETE FROM TB_FUNCAO WHERE ID_FUNCAO = @ID_FUNCAO;
                IF (@@ROWCOUNT= 0)
                    RAISERROR('Erro inesperado, favor
contatar o gerente!', 18, 1)
            END
        ELSE
            IF (@OPR='A') -- ALTERAR
                BEGIN
                    UPDATE TB_FUNCAO SET FUNCAO = @FUNCAO
                    WHERE ID_FUNCAO = @ID_FUNCAO;
                END
            ELSE
                IF (@OPR='I') -- INCLUIR
                    BEGIN
                        INSERT INTO TB_FUNCAO(FUNCAO)
                        VALUES (@FUNCAO);
                    END
            END TRY
        BEGIN CATCH
            IF (@@ROWCOUNT=0)
                RAISERROR('Erro! Nenhuma linha para ser modificado!', 17, 1)
            ELSE
                RAISERROR('Erro inesperado, favor contatar o
gerente!', 18, 1)
            return
        END CATCH
    END;
    EXEC SP_FUNCAO 6, 'JOVEM APRENDIZ', 'I';
-----
CREATE OR ALTER procedure SP_FUNCIONARIO
(
@ID_FUNC integer,
@ID_FUNCAO integer,
@ID_ENDERECHO integer,
@CPF varchar(12),
@email varchar(120),
@TELEFONE varchar(14),
@ID_PESSOA integer,
@OPR char(1))
AS

```

```

@PESSOA integer;
BEGIN
BEGIN TRY
    IF(@OPR='D') -- DELETAR
        BEGIN
            DELETE FROM TB_PERGUNTASECRETA WHERE ID_PERGUNTASECRETA
= (SELECT ID_PERGUNTASECRETA FROM TB_PERGUNTASECRETA WHERE
ID_PERGUNTASECRETA= (SELECT ID_PERGUNTASECRETA FROM
TB_LOGIN WHERE ID_LOGIN = (SELECT ID_LOGIN FROM TB_LOGIN WHERE
ID_PESSOA=(SELECT ID_PESSOA FROM TB_FUNCIONARIO WHERE
ID_FUNC = @ID_FUNC)));
            DELETE FROM TB_LOGIN WHERE ID_PESSOA = (SELECT ID_PESSOA
FROM TB_FUNCIONARIO WHERE
ID_FUNC = @ID_FUNC);

            DELETE FROM TB_ENDERECHO WHERE ID_ENDERECHO= (SELECT
ID_ENDERECHO FROM TB_FUNCIONARIO WHERE
ID_FUNC= @ID_FUNC);
            SELECT ID_PESSOA INTO @PESSOA FROM TB_FUNCIONARIO WHERE
ID_FUNC=@ID_FUNC;

            DELETE FROM TB_FUNCIONARIO WHERE ID_FUNC = @ID_FUNC;

            DELETE FROM TB_PESSOA WHERE ID_PESSOA = @PESSOA;
            IF(@@ROWCOUNT= 0)
                RAISERROR('Erro inesperado, favor contatar o
gerente!',18, 1)
            END
        ELSE
            IF(@OPR='A') -- ALTERAR
                BEGIN
                    UPDATE TB_FUNCIONARIO SET ID_FUNCAO = @ID_FUNCAO,
CPF= @CPF, EMAIL= @EMAIL, TELEFONE= @TELEFONE
                    WHERE ID_FUNC = @ID_FUNC;
                END
        ELSE
            IF(@OPR='I') -- INCLUIR
                BEGIN
                    INSERT INTO TB_FUNCIONARIO(ID_FUNCAO, ID_ENDERECHO, CPF,
EMAIL, TELEFONE, ID_PESSOA)
                    VALUES (@ID_FUNCAO, (SELECT MAX(ID_ENDERECHO) FROM
TB_ENDERECHO), @CPF, @EMAIL, @TELEFONE, (SELECT MAX(ID_PESSOA)
FROM TB_PESSOA));
                END
        END TRY
        BEGIN CATCH
            IF(@@ROWCOUNT=0)
                RAISERROR('Erro! Nenhuma linha para ser modificada!',17, 1)
            ELSE
                RAISERROR('Erro inesperado, favor contatar o
gerente!',18, 1)
            return
        END CATCH
    END;

```

```

-----
CREATE OR ALTER procedure SP_EVENTO
(
@ID_EVENTO integer,
@TITULO varchar(20),
@DATA_EVENTO date,
@DESCRICAO varchar(141),
@HORARIO_INICIO timestamp,
@HORARIO_FINAL timestamp,
@DURACAO varchar(5),
@N_ARTISTAS numeric(3,0),
@TIPO_PAG varchar(20),
@ATIVO NUMERIC(1,0),
@VALOR_EVENTO NUMERIC(10,2),
@ID_ARTISTA_GERAL integer,
@URL_IMG varchar(250),
@OPR char(1))
AS
BEGIN
    BEGIN TRY
        IF(@OPR='D') -- DELETAR
            BEGIN
                DELETE FROM TB_EVENTO WHERE ID_EVENTO =
@ID_EVENTO;

                DELETE FROM TB_EVENTO_ARTISTA WHERE ID_EVENTO =
@ID_EVENTO;

                DELETE FROM TB_EVENTO_INGRESSO WHERE ID_EVENTO =
= @ID_EVENTO;
            END
        ELSE
            IF(@OPR='A') -- ALTERAR
                BEGIN
                    UPDATE TB_EVENTO SET TITULO = @TITULO,
DATA_EVENTO = @DATA_EVENTO, DESCRICAO = @DESCRICAO,
                        HORARIO_INICIO = @HORARIO_INICIO, HORARIO_FINAL
= @HORARIO_FINAL, N_ARTISTAS = @N_ARTISTAS, DURACAO = @DURACAO,
                        TIPO_PAG = @TIPO_PAG, ATIVO = @ATIVO,
VALOR_EVENTO = @VALOR_EVENTO, URL_IMG = @URL_IMG WHERE ID_EVENTO =
@ID_EVENTO;
                END
            ELSE
                IF(@OPR='I') -- INCLUIR
                    BEGIN
                        INSERT INTO TB_EVENTO (TITULO, DATA_EVENTO,
DESCRICAO, HORARIO_INICIO,
                            HORARIO_FINAL,DURACAO
,N_ARTISTAS, TIPO_PAG, ATIVO, VALOR_EVENTO, URL_IMG)
VALUES (@TITULO, @DATA_EVENTO, @DESCRICAO,
@HORARIO_INICIO, @HORARIO_FINAL ,@DURACAO,@N_ARTISTAS,@TIPO_PAG,
@ATIVO, @VALOR_EVENTO, @URL_IMG);
                    END
                END TRY
                BEGIN CATCH

```

```

        RAISERROR('Erro inesperado, favor contatar o
gerente!', 18, 1)
        return
    END CATCH
END;
-----
CREATE OR ALTER procedure SP_DESCONTOS
( SELECT * FROM TB_ARTISTA_GERAL;
SELECT * FROM TB_PESSOA;
SELECT * FROM TB_EVENTO;
SELECT * FROM TB_EVENTO_ARTISTA;
@ID_DESCONTO integer,
@DESCONTO number,
@COD_PROMOCIONAL varchar(),
@ATIVO number,
@ID_CLIENTE integer,
@OPR char)
IS

BEGIN

    IF(@OPR='D') THEN -- DELETAR
        DELETE FROM TB_DESCONTOS WHERE ID_DESCONTO =
@ID_DESCONTO;
    ELSE
        IF(@OPR='A') THEN -- ALTERAR
            UPDATE TB_DESCONTOS SET DESCONTO = @DESCONTO,
COD_PROMOCIONAL=@COD_PROMOCIONAL,ATIVO=@ATIVO
                WHERE ID_DESCONTO = @ID_DESCONTO;
        ELSE
            IF(@OPR='I') THEN -- INCLUIR
                INSERT INTO TB_DESCONTOS(ID_DESCONTO, DESCONTO,
COD_PROMOCIONAL,N_USOS,ATIVO)
                    VALUES (SQ_DESCONTOS.NEXTVAL, @DESCONTO,
@COD_PROMOCIONAL,0, @ATIVO);
            ELSE
                IF(@OPR='U') THEN --USAR DESCONTO

                    UPDATE TB_DESCONTOS SET N_USOS= N_USOS+1 WHERE
ID_DESCONTO= @ID_DESCONTO;
                END IF;
            END IF;
        END IF;
    END IF;

EXCEPTION
    WHEN @EXCEPTION THEN
        RAISE_APPLICATION_ERROR(-20999,'ATENÇÃO! Operação diferente de
I, D, A.', FALSE);
END SP_DESCONTOS;
-----
CREATE OR ALTER procedure SP_INICIAR

```

```

IS

BEGIN

    UPDATE TB_CONTAS SET ATIVO=0 WHERE TB_CONTAS.ID_CONTAS IN
    (SELECT ID_CONTAS FROM TB_CONTAS
     INNER JOIN TB_DATA_CONTA DTC ON DTC.ID_DATA_CONTA=
     TB_CONTAS.ID_CONTAS
     WHERE TB_CONTAS.SITUACAO='Recebida' OR TB_CONTAS.SITUACAO='Paga'
     AND DTC.DATA_CONTA > SYSDATE);

    UPDATE TB_EVENTO SET ATIVO=0 WHERE
    TO_DATE(DATA_EVENTO,'dd/mm/yy')<TO_DATE(SYSDATE,'dd/mm/yy') AND
    ATIVO=1;

    UPDATE TB_EVENTO SET ATIVO=0 WHERE
    TO_DATE(DATA_EVENTO,'dd/mm/yy')<TO_DATE(SYSDATE,'dd/mm/yy') AND
    TO_CHAR(HORARIO_FINAL,'hh24:mi') <TO_CHAR(SYSDATE,'hh24:mi') AND
    CAST(SUBSTR(HORARIO_FINAL,10,5) AS VARCHAR(5)) !='00:00' AND ATIVO=1;

    UPDATE TB_EVENTO SET ATIVO=0 WHERE
    TO_DATE(DATA_EVENTO,'dd/mm/yy')<TO_DATE(SYSDATE,'dd/mm/yy') AND
    TO_CHAR(HORARIO_FINAL+1,'hh24:mi') <TO_CHAR(SYSDATE,'hh24:mi') AND
    CAST(SUBSTR(HORARIO_FINAL,10,5) AS VARCHAR(5)) ='00:00' AND ATIVO=1;

    UPDATE TB_PESSOA SET ATIVO=0 WHERE TB_PESSOA.ID_PESSOA IN
    (SELECT ID_PESSOA FROM TB_ARTISTA_GERAL
     INNER JOIN TB_ARTISTA ON TB_ARTISTA.ID_ARTISTA_GERAL=
     TB_ARTISTA_GERAL.ID_ARTISTA_GERAL
     INNER JOIN TB_AGENTE ON TB_AGENTE.ID_AGENTE=
     TB_ARTISTA.ID_AGENTE
     WHERE TB_AGENTE.ATIVO=0);

    UPDATE TB_EVENTO SET ATIVO=0 WHERE TB_EVENTO.ID_EVENTO IN
    (SELECT TB_EVENTO.ID_EVENTO FROM TB_EVENTO
     INNER JOIN TB_EVENTO_ARTISTA ON TB_EVENTO_ARTISTA.ID_EVENTO=
     TB_EVENTO.ID_EVENTO
     INNER JOIN TB_PESSOA ON TB_PESSOA.ID_PESSOA=
     TB_EVENTO_ARTISTA.ID_PESSOA
     WHERE TB_PESSOA.ATIVO=0);

-----
ALTER procedure [dbo].[SP_LOG]
(
@TIPO_LOG varchar(15),
@Id_FUNC integer,
@TABELA_LOG varchar(15),
@Id_MODIFICADO integer,--ALTERADO, EXCLUIDO, INSERIDO
@DESCRICAO varchar(50),
@OPR char(1))
AS
DECLARE
    @Id_Tipo_Log integer,
    @Id_Tabela_Log integer,
    @Id_Log integer
BEGIN

```

```

BEGIN TRY

    SET @ID_TIPO_LOG=(SELECT ID_TIPO_LOG FROM TB_TIPO_LOG WHERE
TIPO_LOG = @TIPO_LOG);

    SET @ID_TABELA_LOG=(SELECT ID_TABELA_LOG FROM
TB_TABELAS_LOG WHERE TABELA = @TABELA_LOG);

    INSERT INTO TB_LOG (DATA_LOG, ID_TIPO_LOG, ID_FUNC)
VALUES (GETDATE(), @ID_TIPO_LOG, @ID_FUNC);

    SET @ID_LOG= (SELECT MAX(ID_LOG) FROM TB_LOG);

    IF(@OPR='D') -- DELETAR
        BEGIN
            INSERT INTO TB_LOG_EXCLUSAO (ID_LOG,
ID_TABELA_LOG, ID_EXCLUIDO, DESCRICAO)
                VALUES (@ID_LOG, @ID_TABELA_LOG, @ID_MODIFICADO,
@DESCRICAO);

        END
    ELSE
        IF(@OPR='A') -- ALTERAR
            BEGIN
                INSERT INTO TB_LOG_ALTERACAO (ID_LOG,
ID_TABELA_LOG, ID_ALTERADO)
                    VALUES (@ID_LOG, @ID_TABELA_LOG,
@ID_MODIFICADO);
            END
        ELSE
            IF(@OPR='I') -- INCLUIR
                BEGIN
                    INSERT INTO TB_LOG_INSERCAO (ID_LOG,
ID_TABELA_LOG, ID_INSERIDO)
                        VALUES (@ID_LOG, @ID_TABELA_LOG,
@ID_MODIFICADO);
                END
        END TRY
    BEGIN CATCH

        RAISERROR('Erro inesperado, favor contatar o
gerente!', 18, 1)
        return
    END CATCH
END;

-----
CREATE OR ALTER procedure SP_NOVIDADE
(
@ID_BANNER integer,
@URL_IMG varchar(),
@INDEX_BANNER number,
@ATIVO number,
@OPR char)
IS

```

```

BEGIN

    IF (@OPR='D') THEN -- DELETAR
        DELETE FROM TB_BANNER WHERE ID_BANNER =
@ID_BANNER;
    ELSE
        IF (@OPR='A') THEN -- ALTERAR
            UPDATE TB_BANNER SET URL_IMG = @URL_IMG,
INDEX_BANNER=@INDEX_BANNER, ATIVO= @ATIVO
                WHERE ID_BANNER = @ID_BANNER;
        ELSE
            IF (@OPR='I') THEN -- INCLUIR
                INSERT INTO TB_BANNER(ID_BANNER,URL_IMG,
INDEX_BANNER, ATIVO)
                    VALUES (SQ_NOVIDADE.NEXTVAL, @URL_IMG,
@INDEX_BANNER, @ATIVO);
            END IF;
        END IF;
    END IF;

EXCEPTION
    WHEN @EXCEPTION THEN
        RAISE_APPLICATION_ERROR(-20999,'ATENÇÃO! Operação diferente de
I, D, A.', FALSE);
END SP_NOVIDADE;
-----

CREATE OR ALTER procedure SP_CANCELAR_VENDA
(
@ID_VENDA integer,
@MOTIVO varchar(),
@ID_FUNC number,
@OPR char)
IS

@ID_CONTAS integer;
BEGIN

    IF (@OPR='C') THEN -- CANCELAR
        UPDATE TB_VENDA_INGRESSO SET SITUACAO=
'Cancelado' WHERE ID_VENDA= @ID_VENDA;

        UPDATE TB_ASSENTO_CLI SET ATIVO=0 WHERE
ID_EVENTO= (SELECT DISTINCT TB_EVENTO_INGRESSO.ID_EVENTO FROM
TB_VENDA_INGRESSO
            INNER JOIN TB_ITEM_VENDA ITV ON ITV.ID_VENDA=
TB_VENDA_INGRESSO.ID_VENDA
            INNER JOIN TB_EVENTO_INGRESSO ON
TB_EVENTO_INGRESSO.ID_EI = ITV.ID_EI
            INNER JOIN TB_EVENTO EV ON EV.ID_EVENTO =
TB_EVENTO_INGRESSO.ID_EVENTO WHERE TB_VENDA_INGRESSO.ID_VENDA=
@ID_VENDA)
            AND ID_CLIENTE = (SELECT DISTINCT ID_CLIENTE FROM
TB_VENDA_INGRESSO WHERE ID_VENDA= @ID_VENDA);

```

```
        INSERT INTO TB_CANCELAVENDA(ID_CANCELAVENDA,
ID_VENDA,MOTIVO,DATA_CANCELAMENTO, ID_FUNC) VALUES
        (SQ_CANCELAVENDA.NEXTVAL, @ID_VENDA, @MOTIVO,
SYSDATE, @ID_FUNC);

        SELECT ID_CONTAS INTO @ID_CONTAS FROM
TB_DETALHE_CONTA_VENDA WHERE ID_VENDA= @ID_VENDA;

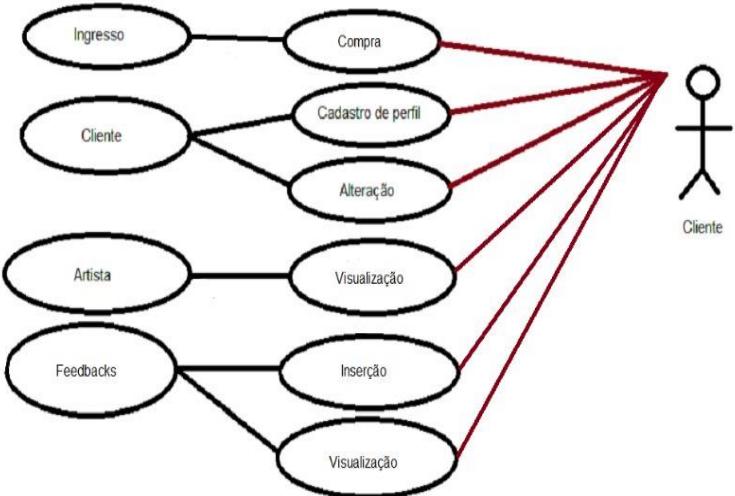
        DELETE FROM TB_DETALHE_CONTA WHERE ID_CONTAS=
@ID_CONTAS;
        DELETE FROM TB_DETALHE_CONTA_VENDA WHERE
ID_CONTAS= @ID_CONTAS;
        DELETE FROM TB_CONTAS WHERE ID_CONTAS =
@ID_CONTAS;

END IF;

-----
EXEC proc_Pessoa (1,'teste','A','I');
select * from TB_PESSOA;
```

3.5 UML

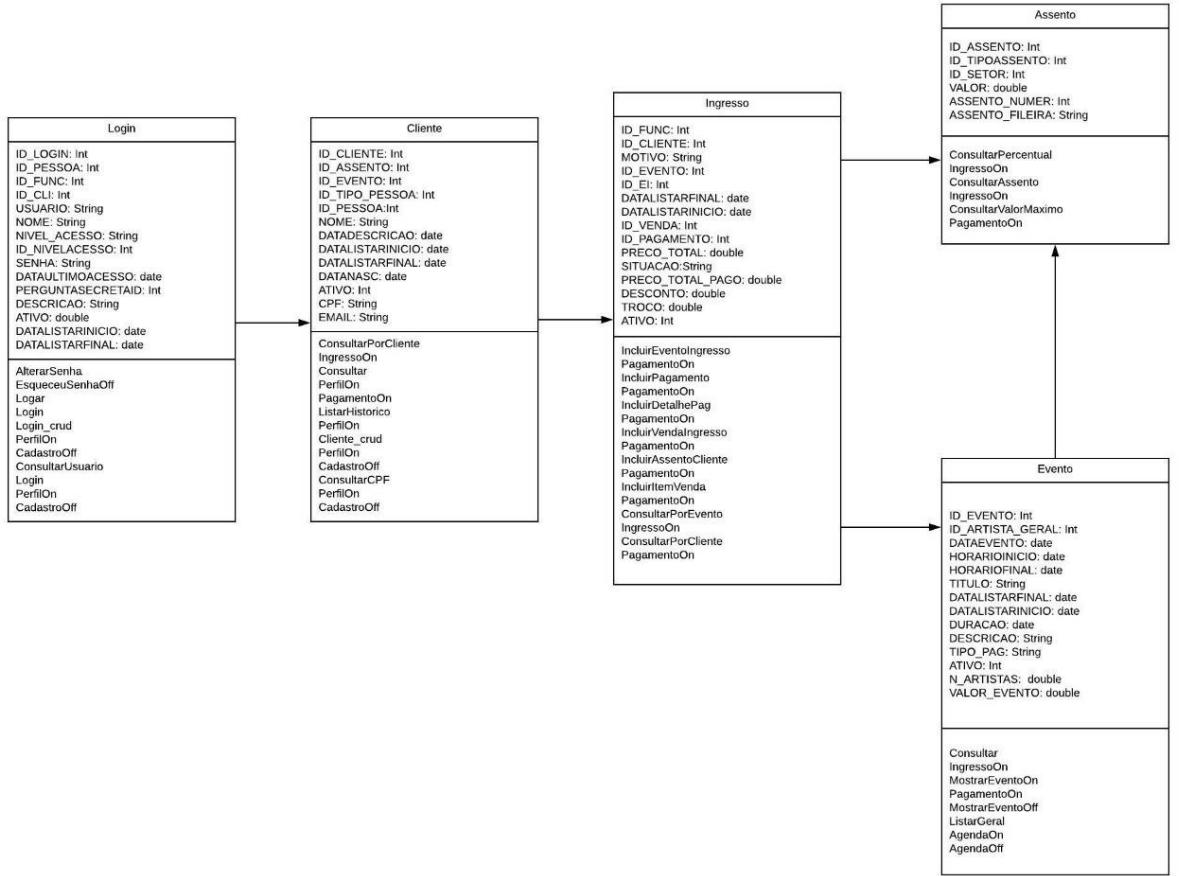
3.5.1 Diagrama de Caso de Uso



3.5.2 Caso de Uso

O cliente se cadastrará no site da casa, podendo assim depois fazendo algumas alterações em seu perfil. O cliente efetuará a compra de ingressos, e poderá visualizar os artistas cadastrado no sistema coorporativo. O cliente poderá inserir um feedback e visualizar outros feedbacks deixados por outros usuários

3.5.3 Diagrama de Classes



3.5.4 Representação das Classes

Artistas

Artista
ID_ARTISTA_GERAL: Int ID_TIPO_PESSOA: Int ID_PESSOA: Int ID_AGENTE: Int ID_ENDERECHO: Int ID_REDESOCIAL: Int DATALISTARFINAL: date DATALISTARINICIAL: date SEXO: String URL_IMG: String TELEFONE: String NOME: String FACEBOOK: String TWITTER: String INSTAGRAM: String EMAIL: String SOCIAL: String DATADECRIACAO: date DATANASC: date ATIVO: Int CPF: String
ListarHistoric MostrarArtistaOn ConsultarNORMAL MostrarArtistaOn ConsultarFIXO MostrarArtistaOn ConsultarPorNome AgendaOn AgendaOff ListarGeral ArtistaOn ArtistaOff

Imagen 89 – Classe Artista

Assento

Assento
ID_ASSENTO: Int
ID_TIPOASSENTO: Int
ID_SETOR: Int
VALOR: Double
ASSENTO_NUMER: Int
ASSENTO_FILEIRA: String
ConsultarPercentual
IngressoOn
ConsultarAssento
IngressoOn
ConsultarValorMaximo
PagamentoOn
ConsultarValorMaximo

Imagen 90 – Classe Assento

Boleto

Boleto
AddBoleto
PagamentoOn

Imagen 91 – Classe Boleto

CEP

CEP
ID_ENDERECHO: Int
CEP: String
NUMERO: double
COMPLEMENTO: String
Consultar
PagamentoOn

Imagen 92 – Classe CEP

Cliente

Cliente	Cliente
ID_CLIENTE: Int ID_ASSENTO: Int ID_EVENTO: Int ID_TIPO_PESSOA: Int ID_PESSOA:Int NOME: String DATADESCRICAOP: date DATALISTARINICIO: date DATALISTARFINAL: date DATANASC: date ATIVO: Int CPF: String EMAIL: String	ConsultarPorCliente IngressoOn Consultar PerfilOn PagamentoOn ListarHistorico PerfilOn Cliente_crud PerfilOn CadastroOff ConsultarCPF PerfilOn CadastroOff ListarGeral ListarHistorico ConsultarCPF Cliente_crud

Imagen 93 – Classe Cliente

Contas

Contas
ID_CONTAS: Int ID_VENDA: Int ID_DATACONTA: Int DATA: date DATALISTARFINAL: date DATALISTARINICIO: date DATA_ENTREGUE: date DATALANCAMENTO: date TIPO_CONTA: String TIPO_DATA: String DESCRICAQ: String DEPARTAMENTO: String FORMA_PAGAMENTO: String SITUACAO: String VALOR_TOTAL: double ATIVO: Int ID_FORMA_PAGAMENTO: Int
IncluirVenda PagamentoOn Contas_crud PagamentoOn

Imagen 94 – Classe Contas

Descontos

Descontos
ID_DESCONTOS: Int ID_CLIENTE: Int DESCONTO: double CODPROMOCIONAL: String N_USOS: double ATIVO: Int
Consultar PagamentoOn Descontos_crud PagamentoOn

Imagen 95 – Classe Descontos

Login	
Login	
ID_LOGIN: Int	
ID_PESSOA: Int	
ID_FUNC: Int	
ID_CLI: Int	
USUARIO: String	
NOME: String	
NIVEL_ACESSO: String	
ID_NIVELACESSO: Int	
SENHA: String	
DATAULTIMOACESSO: date	
PERGUNTASECRETAID: Int	
DESCRICAO: String	
ATIVO: Int	
DATALISTARINICIO: date	
DATALISTARFINAL: date	
AlterarSenha	
EsqueceuSenhaOff	
Logar	
Login	
Login_crud	
PerfilOn	
CadastroOff	
ConsultarUsuario	
Login	
PerfilOn	
CadastroOff	

Imagen 96 – Classe Login

Ingresso

Ingresso

ID_FUNC: Int
 ID_CLIENTE: Int
 MOTIVO: String
 ID_EVENTO: Int
 ID_EI: Int
 DATA LISTAR FINAL: date
 DATA LISTAR INICIO: date
 ID_VENDA: Int
 ID_PAGAMENTO: Int
 PRECO_TOTAL: Double
 SITUACAO: String
 PRECO_TOTAL_PAGO: Double
 DESCONTO: Double
 TROCO: Double
 ATIVO: Int

IncluirEventoIngresso
 PagamentoOn
 IncluirPagamento
 PagamentoOn
 IncluirDetalhePag
 PagamentoOn
 IncluirVendaIngresso
 PagamentoOn
 IncluirAssentoCliente
 PagamentoOn
 IncluirItemVenda
 PagamentoOn
 ConsultarPorEvento
 IngressoOn
 ConsultarPorCliente
 PagamentoOn

Imagen 97 – Classe Ingresso

Evento

Evento
ID_EVENTO: Int ID_ARTISTA_GERAL: Int DATAEVENTO: date HORARIOINICIO: date HORARIOFINAL: date TITULO: String DATALISTARFINAL: date DATALISTARINICIO: date DURACAO: date DESCRICAO: String TIPO_PAG: String ATIVO: Int N_ARTISTAS: Double VALOR_EVENTO: double Consultar IngressoOn MostrarEventoOn PagamentoOn MostrarEventoOff ListarGeral AgendaOn AgendaOff

Imagen 98 – Classe Evento

Feedback

Feedback
ID_FEEDBACK: Int MENSAGEM: String ID_CLIENTE: Int AVALIACAO: String TITULO: String IncluirMensagem FeedBackOn RemoverMensagem FeedBackOn

Imagen 99 – Classe Feedback

Pessoa

Pessoa
ID_PESSOA: Int
NOME: String
ID_TIPO_PESSOA: Int
DATA_NASC: date
DATA_CRIACAO: date
SEXO: String
ATIVO: Int
Pessoa_crud
PerfilOn
CadastroOff

Imagen 100 – Classe Pessoa

ValidadorCPF

ValidadorCPF
CampoCPF
PerfilOn
CadastroOff

Imagen 101 – Classe ValidadorCPF

3.6 Interface

3.6.1 Telas e Navegação



Imagen 37 – Cabeçalho

Cabeçalho: Mostra o nome da casa e todas as páginas em que se pode navegar.



Imagen 38 – Cabeçalho Logado

Cabeçalho Logado: Cabeçalho mostrado quando o usuário loga.



Principais Funcionalidades



Perfil de Usuário

Tenha acesso total aos seus dados e histórico de compras.



Descontos exclusivos!

Quem baixa o app recebe descontos exclusivos na Casa!



Ingresso no QR Code!

Evide filas possuindo o ingresso em seu celular!



Novidades

Com o app, você fica por dentro das últimas novidades!



Imagen 39 – Rodapé

Rodapé: Parte de baixo do site, onde são mostradas outras informações sobre a casa.

**Imagen 40 – Home**

Home: Nessa tela serão mostradas as últimas novidades no banner rotativo.

Nome Completo Digite seu nome completo... <small>(Campo obrigatório)</small>	Usuário Digite seu usuário... <small>(Campo obrigatório)</small>
CPF Digite seu CPF... <small>(Campo obrigatório)</small>	Senha Digite sua senha... <small>(Campo obrigatório)</small>
Data de Nasc. dd/mm/aaaa <small>(Campo obrigatório)</small>	Pergunta Secreta Digite sua pergunta secreta... <small>(Campo obrigatório)</small>
Sexo Masculino <small>(Campo obrigatório)</small>	Resposta Digite sua resposta... <small>(Campo obrigatório)</small>
Email Digite seu email... <small>(opcional)</small>	

Imagen 41 – Cadastro

Cadastro: Nessa tela o usuário que não possui um cadastro, poderá criar um login que se conectará com o site e o mobile.

The screenshot shows the 'Recuperação de Senha' (Password Recovery) page. At the top, there's a header bar with the 'Comedy House' logo and navigation links for 'Artistas', 'Agenda', 'Ajuda', 'Feedbacks', and 'Login'. The main form has a light gray background and is titled 'Recuperação de Senha'. It contains the following fields:

- Usuário:** A dropdown menu showing 'rebeca'.
- Pergunta Secreta:** A dropdown menu showing 'DATA DE NASCIMENTO DO FILHO'.
- Resposta:** An input field labeled 'Digite sua resposta...' with a red 'Verificar' (Check) button to its right.
- Senha Nova:** An input field labeled 'Digite sua nova senha...'.
- Confirmar Senha:** An input field labeled 'Digite novamente sua senha...'.
- Salvar:** A red 'Salvar' (Save) button at the bottom right.

Imagen 42 – Recuperação de senha

Recuperação de Senha: Caso o usuário tenha esquecido ou perdido a senha, ele poderá recuperá-la através da pergunta secreta.

The screenshot shows the 'Login' page. At the top, there's a header bar with the 'Comedy House' logo and navigation links for 'Artistas', 'Agenda', 'Ajuda', 'Feedbacks', and 'Login'. The main form has a light gray background and is titled 'Login'. It contains the following fields:

- Usuário:** An input field with a user icon and placeholder text 'Digite seu usuário...'.
- Senha:** An input field with a lock icon and placeholder text 'Digite sua senha...', accompanied by a small eye icon to toggle visibility.
- Links:** Below the fields, there are two red links: 'Ainda não possui cadastro?' and 'Esqueceu a senha?'.
- Entrar:** A red 'Entrar' (Enter) button at the bottom right.

Imagen 43 – Login

Login: Nessa tela o usuário que já possui cadastro poderá logar no site.

The screenshot shows the 'Artistas' section of the Comedy House website. At the top, there's a red header bar with the 'Comedy House' logo and navigation links for 'Artistas', 'Agenda', 'Ajuda', 'Feedbacks', and 'Login'. Below the header, a banner says 'Conheça nossos Artistas!' (Meet our artists!). A search bar with placeholder text 'Pesquisar...' and a dropdown menu 'Nome' with a magnifying glass icon are on the right. Underneath, there are three artist profiles in cards:

- DANILO GENTILI**: Shows a photo of Danilo Gentili in a suit, holding a mug. Below the photo is his name and a 'Ver Mais' button.
- FABIO PORCHAT**: Shows a photo of Fábio Porchat with an open mouth, looking surprised. Below the photo is his name and a 'Ver Mais' button.
- GIGANTE LEO**: Shows a photo of Gigante Leo holding a microphone. Below the photo is his name and a 'Ver Mais' button.

Imagen 44 – Artistas

Artistas: Nessa tela será possível ver todos os artistas cadastrados no sistema corporativo, também será possível filtrá-los. Quando se clica no botão de “ver mais” do artista específico, o usuário será redirecionado para a tela “MostrarArtista”.

The screenshot shows the 'MostrarArtista' (Show Artist) page for Marco Luque. At the top, it says 'Olá, REBECA SANTOS! Bem vindo(a) à Comedy House!' and has navigation links for 'Artistas', 'Agenda', 'Ajuda', 'Perfil', 'Feedbacks', and 'Sair'. The main title is 'Detalhes' (Details). On the left is a large photo of Marco Luque pointing at the camera. To the right are several input fields with placeholder text:

- Nome Completo:** MARCO LUQUE
- Data de Nascimento:** 28/10/1985
- Facebook:** MarcoLuqueOficial
- Instagram:** (empty field)
- Twitter:** (empty field)

Below this, under 'Empresário (Pessoa Física)', there are fields for 'Nome do contato' (ROGER GOLDEN), 'Telefone/Celular' ((11)93029-3332), and 'Email' (rogerpirata@gmail.com).

Histórico do Artista

Imagen 45 – MostrarArtista1

Imagen 46 – MostrarArtista2

Mostrar Artista: Nessa tela serão mostradas as informações do artista, como também os shows já feitos.

Imagen 47 - Eventos

Eventos: Onde será mostrado todos os eventos ativos cadastrados no sistema corporativo, também será possível filtrá-los. Quando se clica no botão “Ver mais” do evento específico, o usuário será redirecionado para a tela “Mostrar Evento”.

Oi, **REBECA SANTOS**! Bem vindo(a) à **Comedy House!**

[Artistas](#) [Agenda](#) [Ajuda](#) [Perfil](#) [Feedbacks](#) [Sair](#)

Detalhes



JACKSON FIVE
Organismo Pulsante
Marco Luque

Título
JACKSON FIVE

Data do Evento
01/11/2018

Horário
18:00 á 20:00

Descrição
EVENTO APRESENTADO POR MARCO LUQUE, NA PELE DE JACKSON FIVE.MOTOBOM, CACHORRO LOUCO, APESAR DE FEIO, SE VALORIZA CONTANDO HISTÓRIAS DO COTIDI

Artistas



[Comprar Ingressos](#)

Imagen 48 – Mostrar Evento

Mostrar Evento: Onde serão mostradas as informações do evento.

Olá, REBECA SANTOS! Bem vindo(a) à Comedy House!

Artistas Agenda Ajuda Perfil Feedbacks Sair

JACKSON FIVE

Compra de Ingressos

Palco

Assentos Disponíveis: 163

Legend: Simples (white), Duplo (red), Especial (blue), Selecionado (yellow), Ocupado (black with red X)

Número	Fileira	Tipo	Setor	Valor (R\$)
13 e 14	C	Duplo	Ouro	200
10 e 11	C	Duplo	Ouro	200

Total: 2 Assentos R\$ 400,00

[Avançar](#)

Imagen 49 – Comprar Ingresso

Comprar Ingresso: Nessa tela o usuário escolherá seu assento de preferência e verá informações desse assento, como: número do assento, fileira, tipo, setor e valor, e também poderá ver o valor total da sua compra.

Olá, REBECA SANTOS | Bem vindo(a) à Comedy House!

Artistas Agenda Ajuda Perfil Feedbacks Sair

Finalizar Compra

Valor Total: R\$ 50,00

Código Promocional
MOBILE10 [Remover](#)

Se você possui um Voucher, digite o código correspondente e receba descontos!

Desconto
R\$ 5
Porcentual
10,0%

 Crédito  Boleto  Débito

Bandeira
 

Nº de Parcelas
1x ▾

Li e concordo com os [Termos e condições da Comedy House](#).

Termos e Condições de Compra e Venda de Comedy House, com sede na rua Augusta, 1129, capital de São Paulo., doravante denominada simplesmente Comedy House, e, de outro lado, o cliente Comedy House, qualificado no momento da compra dos produtos Comedy House, doravante denominado simplesmente Cliente.

Considerando que a Comedy House realiza venda de produtos e serviços pela internet; Considerando o interesse do Cliente na compra dos produtos

Comprar

Imagen 50 – Pagamento1

Olá, REBECA SANTOS | Bem vindo(a) à Comedy House!

[Artistas](#) [Agenda](#) [Ajuda](#) [Perfil](#) [Feedbacks](#) [Sair](#)

Valor Total: R\$ 50,00

Código Promocional
MOBILE10 [Remover](#)

Se você possui um Voucher, digite o código correspondente e receba descontos!

Desconto
R\$ 5
Porcentual
10,0%

Crédito Boleto Débito

CEP
Digite seu CEP...

Após digitar o CEP, clique no botão do lado, para ser feita a busca do endereço.

Logradouro
Digite seu Logradouro...

Bairro
Digite seu Bairro...

Cidade
Digite sua Cidade...

Estado
AC

Número
Digite o Número...

Complemento(Opcional)
Digite o Complemento...

Li e concordo com os [termos e condições da Comedy House.](#)

Imagen 51 – Pagamento2

Pagamento: Nessa tela o usuário efetuará o pagamento, finalizando sua compra. Caso o usuário possua um código promocional ele poderá inseri-lo. Logo abaixo ele escolherá a forma de pagamento (cartão de crédito, débito e boleto). Caso a forma de pagamento seja cartão o usuário terá que informar a bandeira e o número de parcelas, e então terá que concordar com os termos de uso da casa de show e finalizar a compra.

The screenshot shows the 'Ajuda' (Help) section of the Comedy House website. At the top, there is a red header bar with the 'Comedy House' logo on the left and navigation links for 'Artistas', 'Agenda', 'Ajuda' (which is underlined), 'Feedbacks', and 'Login' on the right. Below the header, the page title 'Dúvidas mais frequentes' (Most frequent questions) is centered. A list of seven frequently asked questions is displayed in a grid format, each with a '+' sign to expand the answer:

- Como se dá o valor de um ingresso?
- Menores de idade podem assistir a um show?
- Qual a duração de um show?
- Quantos ingressos posso comprar?
- Comprei meu ingresso online, onde retiro?
- Sobre o preço dos ingressos...
- Em quantas vezes posso parcelar o pagamento?

Imagen 52 – Ajuda1

The screenshot shows the 'Ajuda' (Help) section of the Comedy House website. At the top, there is a red header bar with the 'Comedy House' logo on the left and navigation links for 'Artistas', 'Agenda', 'Ajuda' (which is underlined), 'Feedbacks', and 'Login' on the right. Below the header, the page title 'Dúvidas mais frequentes' (Most frequent questions) is centered. A list of two frequently asked questions is displayed in a grid format, each with a '+' sign to expand the answer:

- Como me alimento na Comedy House?
- Não consegui imprimir meu comprovante, o que faço?

Below the questions, a message says 'Não tirou sua dúvida?' (Didn't you get your answer?). There is a form for contacting support, including fields for Name, Email, Phone, and Message, along with a 'Enviar' (Send) button.

Imagen 53 – Ajuda2

Ajuda: Aqui serão mostradas dúvidas frequentes que outros usuários possuem, e também terá informações do contato da casa de show, caso a dúvida do usuário não tenha sido esclarecida.

The screenshot shows the 'Feedbacks' section of the Comedy House website. At the top, there's a navigation bar with links for 'Artistas', 'Agenda', 'Ajuda', 'Feedbacks' (which is underlined), and 'Login'. Below the navigation, the word 'Feedbacks' is centered. A message encourages users to leave reviews: 'Assistiu algum show nosso? Gostou? Deixe uma mensagem de avaliação, isso é extremamente importante para nós!'. A link 'Faça Login para deixar uma mensagem' is provided. Two feedback cards are displayed:

- MUITO BOM!**
Adorei! Nunca tinha ido a um show antes, essa foi minha primeira experiência e gostei!

REBECA SANTOS
25/10/2018 19:00
- Achei meio salgado os preços dos eventos!

WILSON SOARES DE ABREU
25/10/2018 23:33

Imagen 54 – Feedback

Feedback: Nessa tela será mostrado todos os comentários dos usuários cadastrados no nosso site, caso o comentário seja positivo ele aparecerá na cor verde, caso ele seja um comentário intermediário aparecerá na cor amarela, e caso seja negativo aparecerá na cor vermelha.

The screenshot shows the 'Feedbacks' section for a logged-in user named REBECA SANTOS. At the top, there's a navigation bar with links for 'Artistas', 'Agenda', 'Ajuda', 'Perfil', 'Feedbacks' (underlined), and 'Sair'. Below the navigation, the word 'Feedbacks' is centered. A message encourages users to leave reviews: 'Assistiu algum show nosso? Gostou? Deixe uma mensagem de avaliação, isso é extremamente importante para nós!'. The feedback form includes fields for 'Título (opcional)' (optional title) and 'Mensagem' (message). It also features a star rating system and contact information: '(11)2392-2899 / (11)8976-3200' and 'comedy.house.oficial@gmail.com'. Below the form, a feedback card is shown:

MUITO BOM!
Adorei! Nunca tinha ido a um show antes, essa foi minha primeira experiência e gostei!

REBECA SANTOS
25/10/2018 19:00

Imagen 55 – Feedback Logado

Feedback Logado: Aqui um usuário logado no site poderá escrever e postar uma análise de sua experiência com os serviços da casa de show.

Histórico

De Até

Imagen 56 – Perfil

Histórico

De Até

• Dia 30/10/2018 07:29
• Dia 30/10/2018 02:15
• Dia 30/10/2018 02:14
• Dia 30/10/2018 02:13
• Dia 30/10/2018 02:11
• Dia 30/10/2018 02:08
• Dia 30/10/2018 02:07
• Dia 30/10/2018 02:05
• Dia 30/10/2018 02:01
• Dia 30/10/2018 01:57
• Dia 29/10/2018 08:03
• Dia 28/10/2018 21:22
• Dia 27/10/2018 18:31
• Dia 27/10/2018 18:29
• Dia 27/10/2018 03:21
• Dia 27/10/2018 02:57
• Dia 23/09/2018 19:19

Imagen 57 – Perfil2

Perfil: Onde serão mostradas as informações do usuário que está logado, também com opção de alterá-las e também sendo possível alterar o login.

4 SISTEMA PARA DISPOSITIVOS MÓVEIS (Android)

4.1 Escopo Geral do Sistema

4.1.1 Escopo do Sistema

O escopo do sistema baseia-se na utilização do QR Code como um ingresso virtual junto com o smartphone, pois ela é a característica principal do aplicativo. O cliente precisa levar apenas seu smartphone mostrando o QR Code correspondente e colocar frente à máquina para validar sua entrada num evento.

4.1.2 Funções do Sistema

A principal função do aplicativo é a utilização do QR Code como ingresso virtual, substituindo assim um ingresso físico.

Um cliente que efetuou sua compra com antecedência na casa ou pelo website pode utilizar seu mesmo perfil do website no mobile e checar, através de seu histórico, o QR Code correspondente à venda, e assim apresentá-lo na máquina para validar sua entrada.

4.2 Dados do Sistema

4.2.1 Dados Armazenados

<u>NOME DO ATRIBUTO</u>	<u>TIPO DO ATRIBUTO</u>	<u>DESCRICAÇÃO DO ATRIBUTO</u>
NOME CAMPO	TIPO, AUTO NUM., PK	Descrição do Campo.
NOME CAMPO	TIPO, FK	Descrição do Campo.

Tabela 3 – Dicionário de dados

4.3 Telas

4.3.1 Navegação das Telas

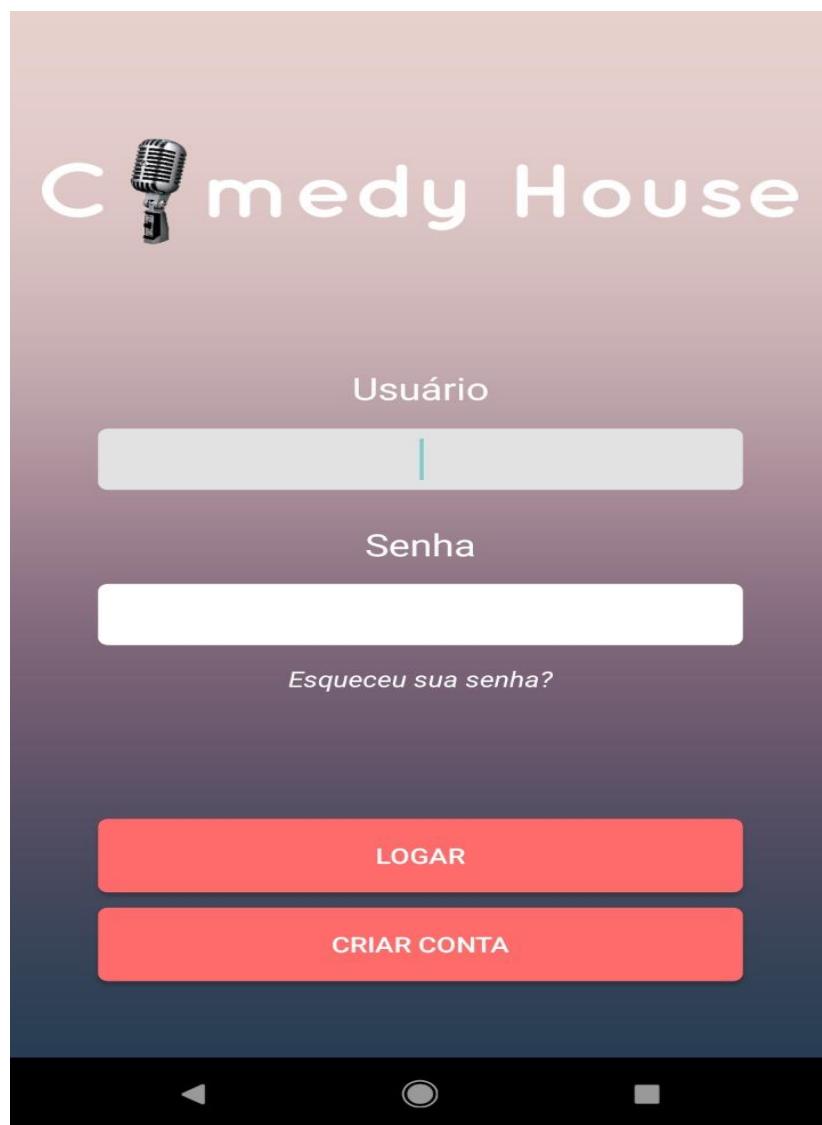


Imagen 58 – Login

Login: Essa tela é onde o usuário poderá fazer login na sua conta do aplicativo mólide, para isso ele deverá preencher os seguintes campos: Usuário e senha. Logo após isso deverá clicar no botão logar, para entrar em sua conta. Nessa tela também contém o botão “Criar conta” e o botão “esqueceu sua senha”.

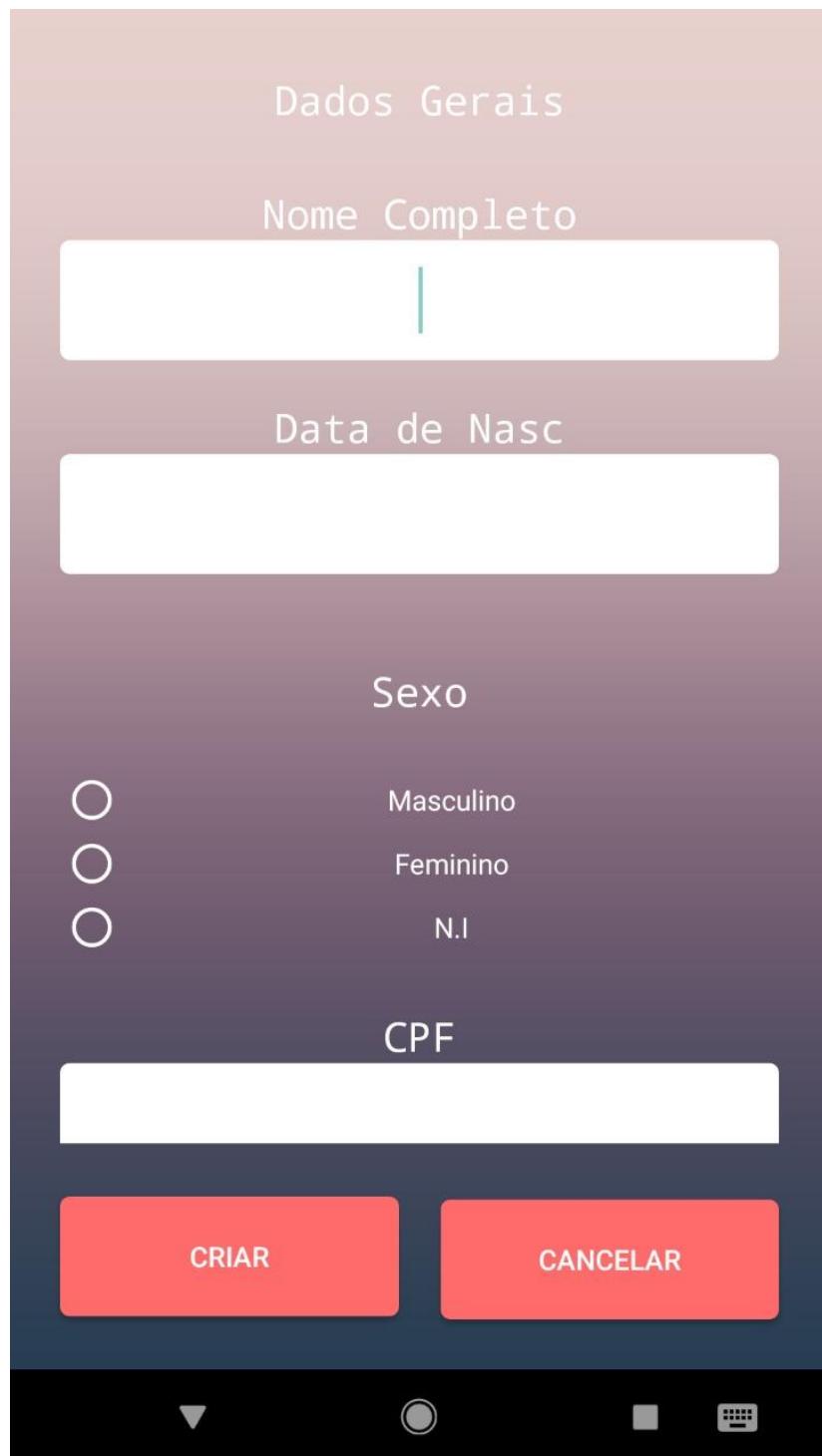


Imagen 59 – Criar Conta 1



Imagen 60 – Criar Conta 2

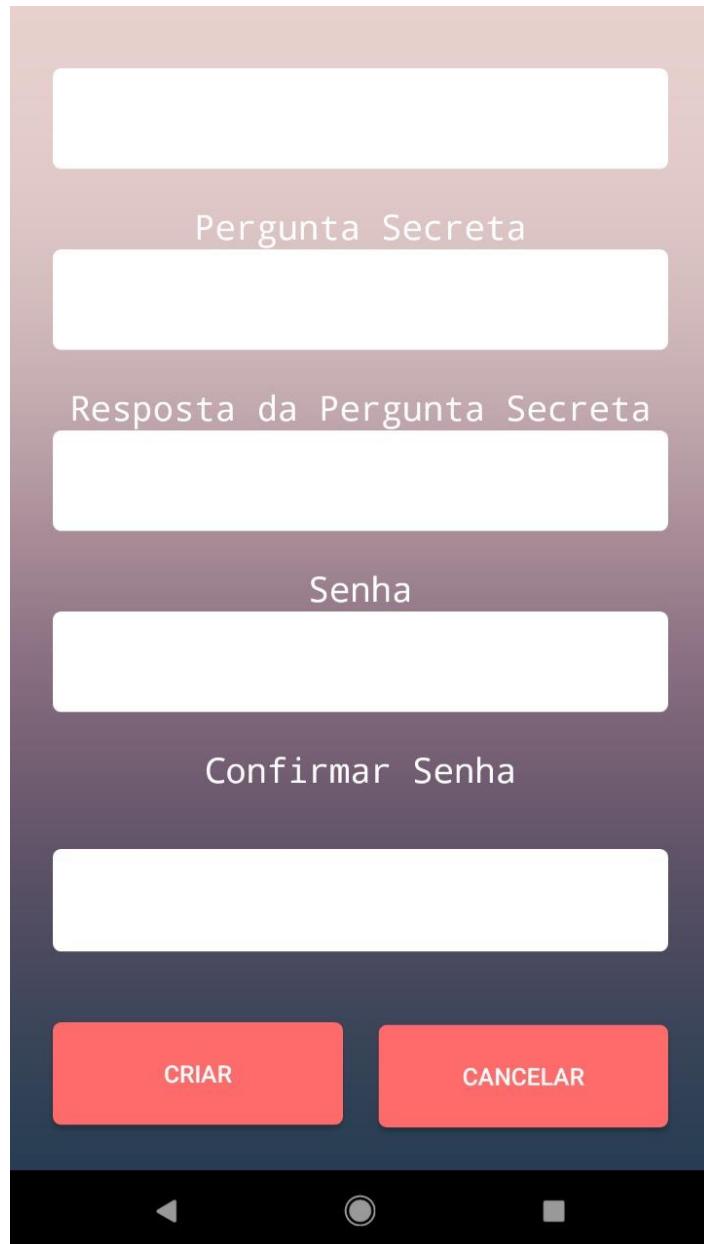


Imagen 61 – Criar Conta 3

Criar conta: Caso a pessoa ainda não possua uma conta ela terá que ir até tela “Criar Conta”, para fazer seu cadastro. Para isso ela deverá preencher os seguintes campos: Nome completo, Data de Nascimento, sexo, CPF, email, usuário, pergunta secreta, resposta da pergunta secreta, senha e o campo confirmar senha. Após isso ela clica no botão criar e sua conta já está criada.



Imagen 62 – Pergunta secreta

Pergunta secreta: Na tela e pergunta secreta é onde a pessoa já cadastrada no aplicativo poderá recuperar sua senha, caso tenha perdido. Para recuperar a senha ela deverá preencher os campos: usuário, pergunta secreta (pergunta que ela cadastrou na hora da criação da conta) e a resposta para essa pergunta, senha e confirmar senha, logo após ela deverá apertar no botão criar para criar uma nova senha.

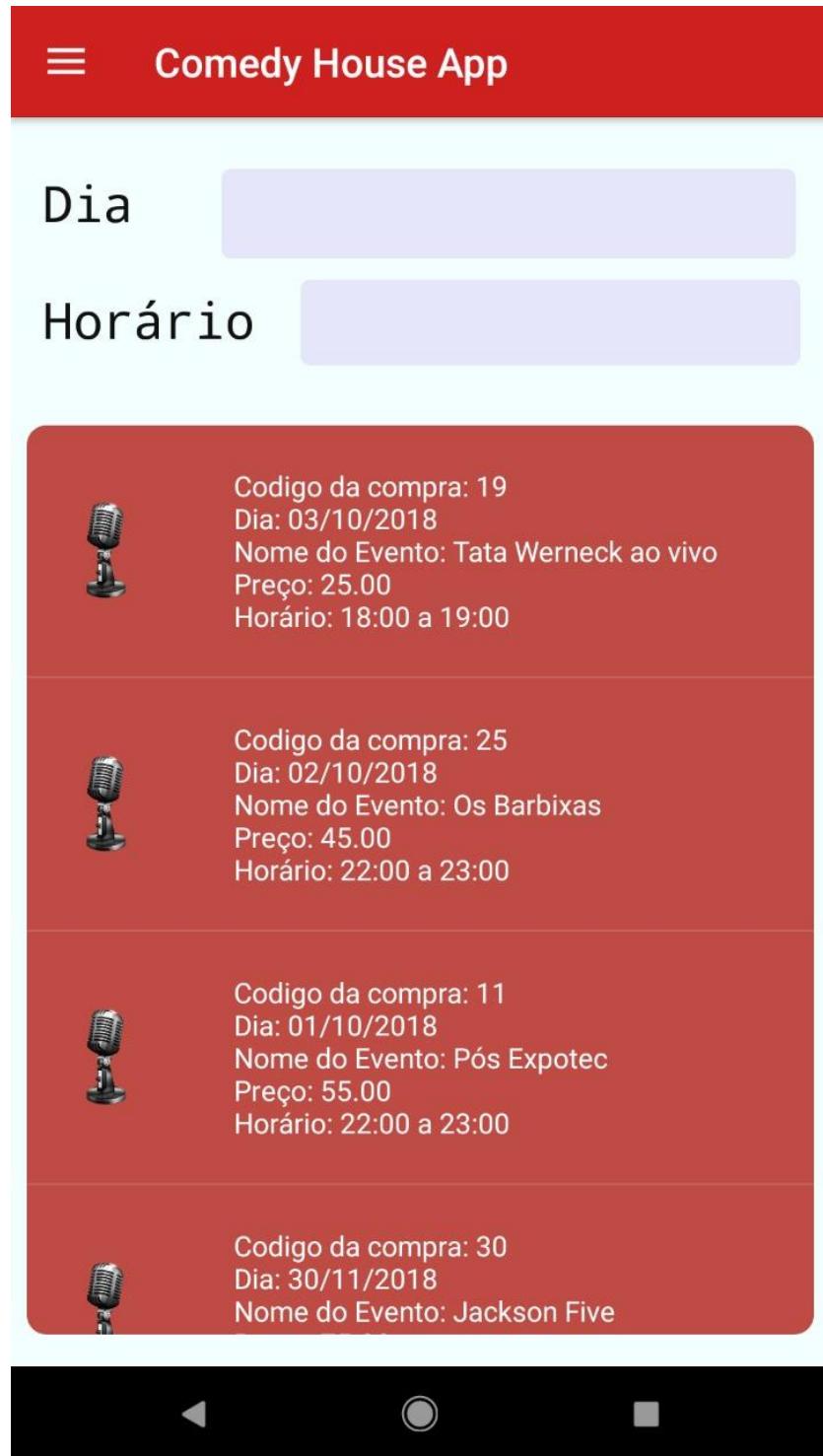


Imagen 63 – Histórico

Histórico: Na tela do histórico o usuário poderá consultar todas as compras feitas por ele na casa de show, podendo filtrar pelo dia e pelo horário. Caso ele clique em cima de uma compra ele será direcionado para a tela do QRCode.

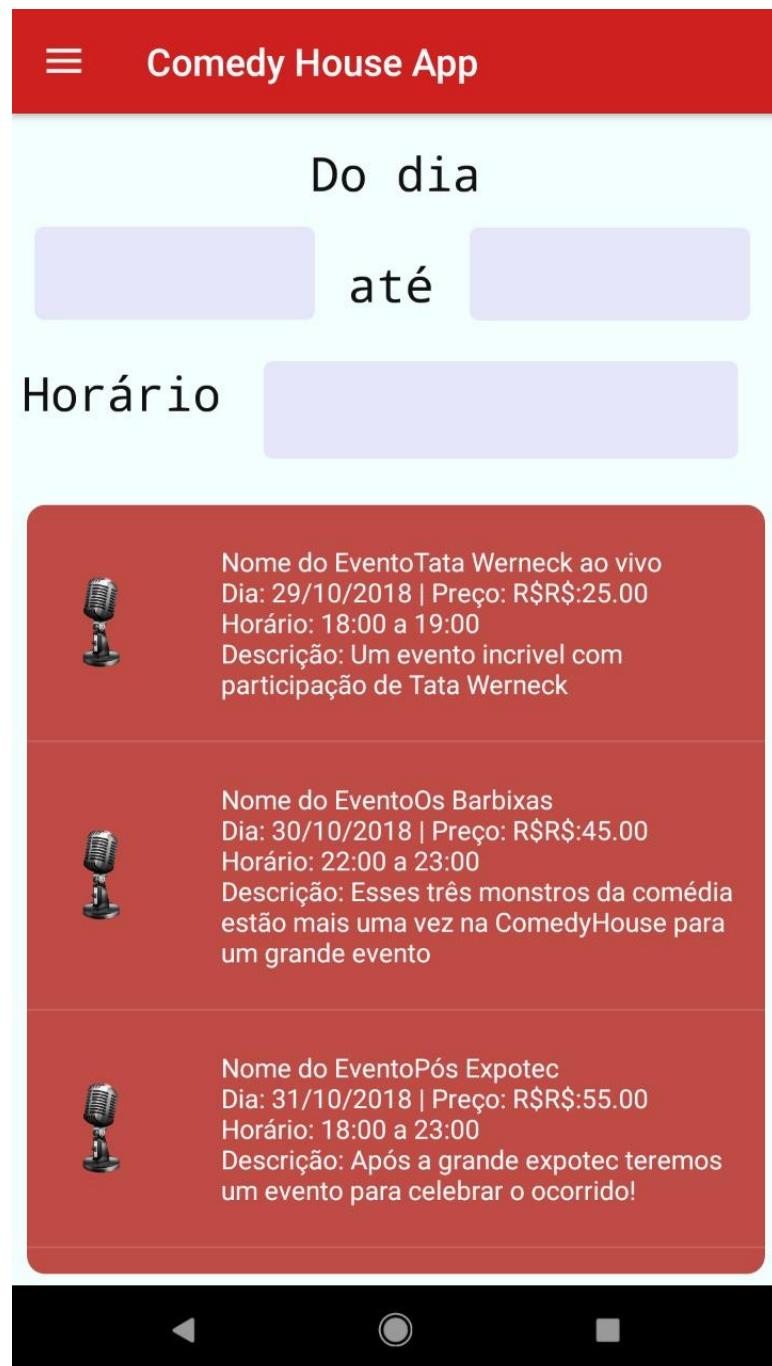


Imagen 64 – Consultar Evento

Consultar Evento: Nessa tela todos os eventos ativos da casa poderá ser consultado, podendo filtrar pelo dia e pelo horário, caso a pessoa clique em cima de um evento, ela poderá consultar todas as informações desse evento detalhadamente.

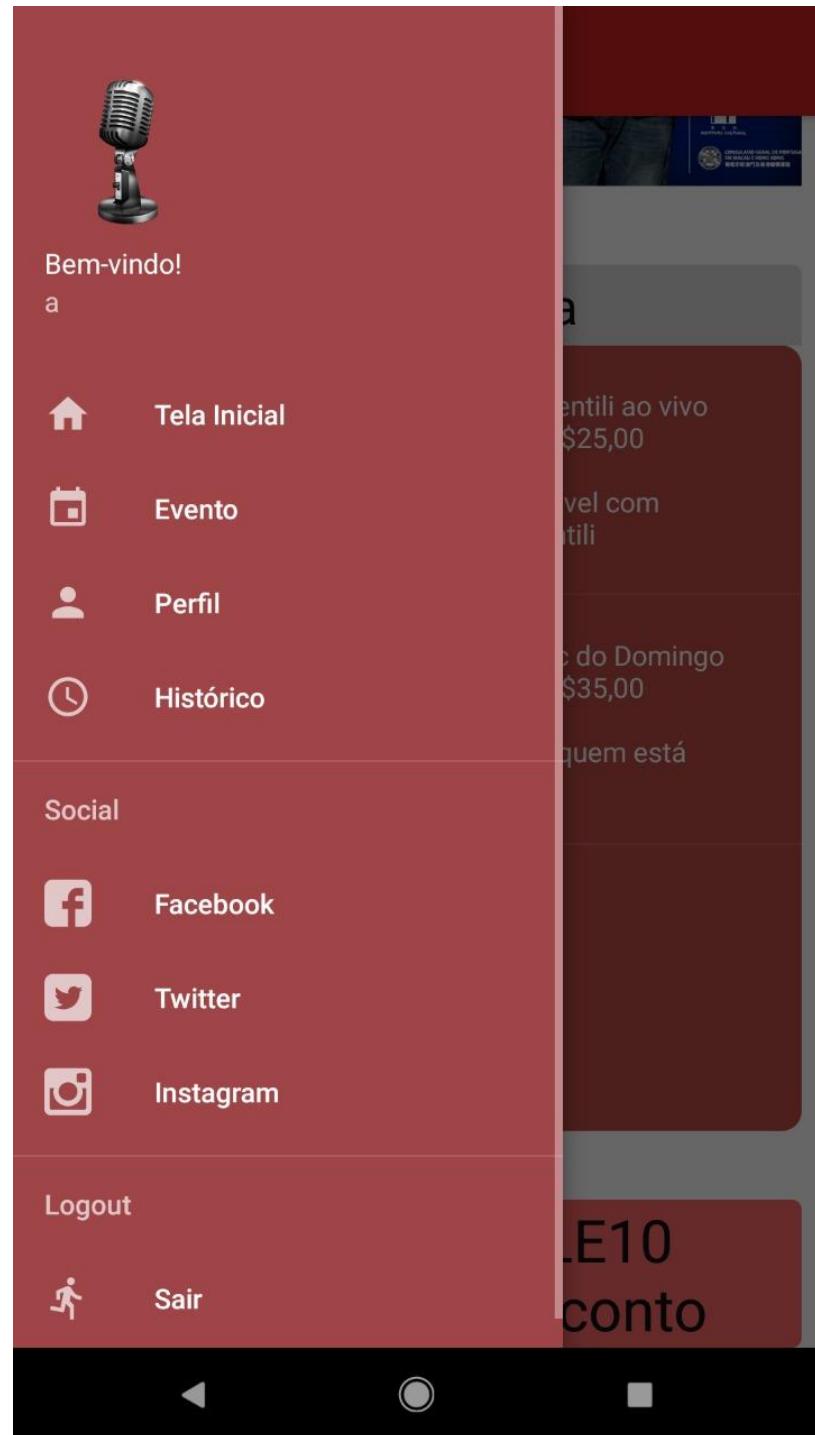


Imagen 65 – Hub 1

The screenshot shows the interface of the Comedy House App. At the top, there's a red header bar with a menu icon (three horizontal lines) and the text "Comedy House App". Below this is a promotional banner for a stand-up comedy show. The banner features a photo of a man (Jorge Serafim) on stage with his mouth open, holding a microphone and a tablet. The text on the banner reads: "STAND-UP COMEDY COM JORGE SERAFIM", "13 DE NOVEMBRO", "20H00", "AUDITÓRIO DR. STANLEY HO", and "CONSULADO-GERAL DE PORTUGAL EM MACAU". A red circular badge says "ENTRADA LIVRE!". Logos for "IPOR" and "CONSULADO-GERAL DE PORTUGAL EM MACAU" are also present. Below the banner, there's a section titled "Eventos do dia" (Events of the day) with two entries. Each entry includes a small microphone icon and details about the event: name, date, price, time, and description. The background of the app interface is white.

Eventos do dia

Nome do Evento: Danilo Gentili ao vivo
Dia: 28/10/2018 | Preço: R\$25,00
Horário: 18:00 a 19:00
Descrição: Um evento incrivel com participação de Danilo Gentili

Nome do Evento: Open Mic do Domingo
Dia: 28/11/2018 | Preço: R\$35,00
Horário: 19:00 a 20:30
Descrição: Open Mic para quem está começando a fazer piadas

Imagen 66 – Hub 2

The screenshot shows the 'Hub' screen of the Comedy House App. At the top, there's a red header bar with a menu icon (three horizontal lines) and the text 'Comedy House App'. Below the header, there's a banner for 'AUDITORIO DR. STANLEY HO' and 'CONSULADO-GERAL DE PORTUGAL EM MACAU', featuring a photo of a person's legs and feet. The main content area has a light gray header 'Eventos do dia'. Below it, there are two event cards on a red background. Each card features a microphone icon and details about an event:

- Nome do Evento:** Danilo Gentili ao vivo
Dia: 28/10/2018 | **Preço:** R\$25,00
Horário: 18:00 a 19:00
Descrição: Um evento incrível com participação de Danilo Gentili
- Nome do Evento:** Open Mic do Domingo
Dia: 28/11/2018 | **Preço:** R\$35,00
Horário: 19:00 a 20:30
Descrição: Open Mic para quem está começando a fazer piadas

At the bottom of the screen, there's a red button with white text: 'CUPOM: MOBILE10 para 10% de desconto'. Below the button is a black navigation bar with three icons: a left arrow, a circle, and a square.

Imagen 67 – Hub 3

Hub: Logo após logar o usuário será direcionado para a tela do hub, onde ele poderá ver um banner com as principais novidades da casa de show, poderá ver todos os eventos do dia, e os cupons de descontos da casa.

Perfil

.....
a

Data de Nasc
1

Sexo
Masculino

CPF
1

Email
a

Informações de Login

Usuário
a

Senha
•



Imagen 68 – Perfil

Perfil: Na tela do perfil o usuário poderá consultar todos os dados da sua conta, como: O usuário a senha etc.

Este QRcode
corresponde
ao seu ingresso.



Imagen 69 – QR Code

QRCode: A tela do QRCode mostrará um QRCode que será gerado com a compra do ingresso. O mesmo poderá ser utilizado na casa de shows, como um comprovante de compra.

Nome	Função
Jose Antonio	Programador
Viviane Lima	Programadora

Tabela 4 – Funções dos participantes novos

5 CONCLUSÃO

Este projeto proporcionou a equipe uma grande experiência de trabalho em equipe e cooperação mútua, no qual todos os integrantes colaboraram uns com os outros e foram muito importantes para a realização do trabalho. Além disso, serviu de base para cada componente ter uma noção de como funciona o mercado de trabalho e suas exigências, como, a falta de tempo, indisponibilidade de reuniões em grupo, pedidos do cliente, etc.

O grupo, passando pelas grandes dificuldades que surgiram conforme o projeto foi sendo realizado, obteve resultados favoráveis e isso proporcionou a conclusão do projeto. A equipe poderia ter tido um desempenho melhor, caso tivesse melhores condições para o desenvolvimento.

REFERÊNCIAS

OLIVEIRA, Pedro Carlos. Sistemas Financeiros. São Paulo: Alínea, 2000.

CHIAVO, Adalberto. Gestão de Recursos. São Paulo: Érica, 1999.

IBGE. Pesquisa Mensal de Emprego. Disponível em <http://ibge.gov.br>. Acesso em 26 de maio de 2006.

APÊNDICE A
FICHA CADASTRAL

**INSTITUTO TÉCNICO DE BARUERI
BRASÍLIO FLORES DE AZEVEDO
CURSO TÉCNICO EM INFORMÁTICA**

Trabalho de conclusão de curso realizado no Instituto de Tecnologia do Instituto Técnico de Barueri Brasílio Flores de Azevedo.

FICHA CADASTRAL

Nome:	_____
RM	_____
Turma / ano-semestre	_____
Data Nascimento	_____
Endereço:	_____
Bairro:	_____
Cidade/UF	_____
Telefone:	_____
E-mail:	_____
Profissão:	_____

ANEXO A
MIDIA COM CÓDIGO FONTE

Colar o envelope aqui
Com o CD ou DVD

**INSTITUTO TÉCNICO DE BARUERI
BRASÍLIO FLORES DE AZEVEDO**

Biblioteca Digital – Termo de Autorização

**Trabalho de Conclusão de Curso
Curso Técnico em Informática**

Nome do Aluno:

RM.: Rg:

Turma:

Título do Trabalho: Casa de Shows de Stand-UP

Autorizo a FIEB – Fundação Instituto de Educação de Barueri, Mantenedora do Instituto Técnico de Barueri Brasílio Flores de Azevedo, estabelecido à Av. Grupo Bandeirante, 138, Jd. Belval, na cidade de Barueri, Estado de São Paulo, inscrita no CNPJ nº 65.700.239/0001-10, a disponibilizar para uso educacional e acadêmico, por prazo indeterminado, gratuitamente e sem o pagamento de qualquer contraprestação, o texto integral da obra supramencionada e o código fonte dos programas, scripts de bancos de dados, layouts de telas referentes aos sistemas desenvolvidos, de minha autoria, em sua página eletrônica mantida na Internet, Biblioteca e Intranet localizada em sua rede local, a título de divulgação da produção científica, podendo a obra ser lida e/ou impressa por todos, a partir desta data.

Barueri, SP 27 / 11 / 2018

Assinatura do (a) aluno (a)