## ‘Palavras-Chave: XXXX, YYYYY, ZZZZZ

# **INTRODUÇÃO**

O sistema de aplicação de barbearia, será um sistema de agendamentos de horário, também um sistema de opções de corte, com álbum de cortes realizados anteriormente para servir de amostra de como os cortes no local funcionam, cadastro de clientes e um sistema de pagamento, para caso o cliente escolha pagar ao realizar o agendamento.



# **FINALIDADE, ESCOPO E OBJETIVO**

Será desenvolvido um sistema para automatização do dos processos de agendamento de horários, a automatização de pagamentos, será desenvolvido um sistema para cadastro de clientes, também mantendo o compromisso de ajudar pessoas de baixa renda e que tem o compromisso com o correto segundo a sociedade e com isso poder contribuir com o trabalho desses profissionais.

**Objetivo Geral**

Esse sistema tem como objetivo automatizar os serviços realizados na barbearia, como agendamentos, cadastros de clientes e visualização dos serviços oferecidos pela barbearia. E o posicionamento da barbearia.

**Objetivo Especifico**

* Realizar agendamentos de serviços
* Realizar o cadastro de clientes
* Realizar o cadastro de funcionários
* Visualização dos serviços oferecidos pela barbearia.

# **DESENVOLVIMENTO**

Criação de Banco De Dados utilizando o MySql Workbench e linguagem SQL, o back end/aplicativo desktop será desenvolvido na linguagem /C# utilizando o framework (Visual Studio). Para o front end/aplicativo será utilizado: HTML, CSS, PHP, javascript, Jquery, bootstrap com o framework (dreamweaver)

# **SÍNTESE**

Este sistema tem por fim otimizar e automatizar os agendamentos de serviços e a visualização dos de produtos na barbearia, além de influenciar na divulgação da barbearia. A partir do cadastro, o cliente poderá realizar agendamentos de serviços na barbearia, visualizar os seus agendamentos. Os funcionários poderão cadastrar clientes, cadastrar fornecedores e realizar pedidos de produtos pelo sistema de vendas realizadas na barbearia.

# **BIBLIOGRAFIA**

# **Turma/ Curso: 76/2022**

# **Componentes do Grupo:**

Vinícius De Souza Silva

Bruno Assunção

Fellipy Pereira Da Rosa

# **Instrutor Responsável: Márcia Cristina Pereira Fortuna**

# **Supervisão: Priscilla Mendes de Oliveira**

# **ANEXOS**

# **REQUISITOS FUNCIONAIS**

|  |  |  |
| --- | --- | --- |
| RF001 – Cadastro de Cliente | | |
| ENTRADA | PROCESSO | SAÍDA |
| Codclie, nome, cpf, telefone, e-mail, login, senha e sexo. | Haverá um campo no banco de dados denominado de ‘cod\_clie’ que será gerado automaticamente para cada cliente, e não haverá repetições. Nenhum dos campos poderá ficar vazio, caso fique será emitido uma mensagem de erro. | Será armazenado esses dados no banco de dados para que seja efetuado o cadastro do cliente. E mostrará a tela ao usuário que o cadastro foi efetuado. |

|  |  |  |
| --- | --- | --- |
| RF002 – Agendamento | | |
| ENTRADA | PROCESSO | SAÍDA |
| Codagendamento, Tipo de serviço, data e hora | O sistema verificará se a data e horário está disponível para realizar o serviço. Caso esteja disponível ele fará o agendamento. O sistema irá verificar também qual o nível do usuário que está efetuando o cadastro, caso seja um funcionário será solicitado os dados do cliente como; Nome, endereço, telefone e cpf para efetuar o cadastro. | Será emitido um comprovante de agendamento ao usuário e será armazenado no banco de dados da agenda o horário agendado. |

|  |  |  |
| --- | --- | --- |
| RF003 – Serviços | | |
| ENTRADA | PROCESSO | SAÍDA |
| Codservico, nomeserv e preçoserv. | Será armazenado esses dados no banco de dados, para que o cliente o consiga realizar o agendamento com base no serviço. | Será mostrado aos usuários os serviços disponíveis na barbearia. |

|  |  |  |
| --- | --- | --- |
| RF004 – Categoria | | |
| ENTRADA | PROCESSO | SAÍDA |
| Codcateg e nomecateg | Será armazenado esses dados no banco de dados, para que o cliente o consiga realizar o cadastro de novos serviços. | Será possível realizar busca de serviços e agendamentos com base na categoria. |

|  |  |  |
| --- | --- | --- |
| RF005 – Cadastro de Funcionários | | |
| ENTRADA | PROCESSO | SAÍDA |
| Nome, endereço, e-mail, cpf e telefone, sexo, login e senha | Será armazenado esses dados no banco de dados, para que os funcionários tenham acesso ao sistema. | Será armazenado esses dados e ao fim do cadastro irá disponibilizar o acesso a áreas restritas no sistema |

|  |  |  |
| --- | --- | --- |
| RF006 – tela de busca | | |
| ENTRADA | PROCESSO | SAÍDA |
| Data do agendamento  Serviço | Será armazenado esses dados no banco de dados, para que os funcionários tenham acesso ao sistema. | Exibira no sistema os serviços feitos e que estão marcados. |

|  |  |  |
| --- | --- | --- |
| RF007 – Tela de login | | |
| ENTRADA | PROCESSO | SAÍDA |
| Login e senha | Depois de inserido no formulário, deverá ser validado a partir do banco de dados. | Estará disponível para fazer o que o usuário estará disposto para fazer. |

# **REGRA DE NEGÓCIO (Define como um requisito funcional se realizará)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | RN001 | | |
| **‘Nome** | CADASTRO DE CLIENTE | | |
| **Data de criação** | 25/08/2022 | **Autor** | Vinícius |
| **Data da última alteração** | 25/10/2022 | **Autor** | BRUNO |
| **Versão** | 1.0 | **Dependências** | RF001 |
| **Descrição** | NÃO PODERÁ DEIXAR NENHUM CAMPO VAZIO, E QUANDO O CADASTRO DO CLIENTE FOR EFETUADO COM SUCESSO APARECERÁ UMA MENSAGEM DE CONFIRMAÇÃO. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | RN002 | | |
| **‘Nome** | AGENDAMENTO | | |
| **Data de criação** | 26/08/2022 | **Autor** | Vinicius |
| **Data da última alteração** | 26/10/2022 | **Autor** | BRUNO |
| **Versão** | 1.0 | **Dependências** | RF002 |
| **Descrição** | DEVERÁ CONFIRMAR TODOS OS CAMPOS PARA O AGENDAMENTO FOR CONCLUIDO, QUANDO FOR CONCLUÍDO IRÁ ARMAZENAR NO BANCO DE DADOS PARA O FUNCIONÁRIO E IRÁ DISPARAR UM E-MAIL/SMS PARA O CLIENTE. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | RN003 | | |
| **‘Nome** | LOGIN DO USUÁRIO | | |
| **Data de criação** | 26/08/2022 | **Autor** | Vinicius |
| **Data da última alteração** | 26/10/2022 | **Autor** | BRUNO |
| **Versão** | 1.0 | **Dependências** | RF007 |
| **Descrição** | DEVERÁ ATENDER AO CADASTRO FEITO NO SISTEMA PARA TER O ACESSO. SE CONFIRMADO, IRÁ TODAS AS FUNÇÕES DO SISTEMA. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | RN004 | | |
| **‘Nome** | CADASTRO DE SERVIÇOS | | |
| **Data de criação** | 26/08/2022 | **Autor** | Vinicius |
| **Data da última alteração** | 26/10/2022 | **Autor** | BRUNO |
| **Versão** | 1.0 | **Dependências** | RF003 |
| **Descrição** | NÃO PODERÁ DEIXAR NENHUM CAMPO VAZIO, E QUANDO O CADASTRO DE SERVÇO FOR EFETUADO COM SUCESSO APARECERÁ UMA MENSAGEM DE CONFIRMAÇÃO. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | RN005 | | |
| **‘Nome** | CADASTRO DE CATEGORIAS | | |
| **Data de criação** | 26/08/2022 | **Autor** | Vinicius |
| **Data da última alteração** | 26/10/2022 | **Autor** | BRUNO |
| **Versão** | 1.0 | **Dependências** | RF004 |
| **Descrição** | NÃO PODERÁ DEIXAR O NOME DA CATEGORIA VAZIA, E QUANDO O CADASTRO DA CATEGORIA FOR EFETUADO COM SUCESSO APARECERÁ UMA MENSAGEM DE CONFIRMAÇÃO. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | RN006 | | |
| **‘Nome** | CADASTRO DE FUNCIONÁRIOS | | |
| **Data de criação** | 26/08/2022 | **Autor** | Vinicius |
| **Data da última alteração** | 26/10/2022 | **Autor** | BRUNO |
| **Versão** | 1.0 | **Dependências** | RF005 |
| **Descrição** | NÃO PODERÁ DEIXAR NENHUM CAMPO VAZIO, E QUANDO O CADASTRO DO FUNCIONÁRIO FOR EFETUADO COM SUCESSO APARECERÁ UMA MENSAGEM DE CONFIRMAÇÃO. | | |

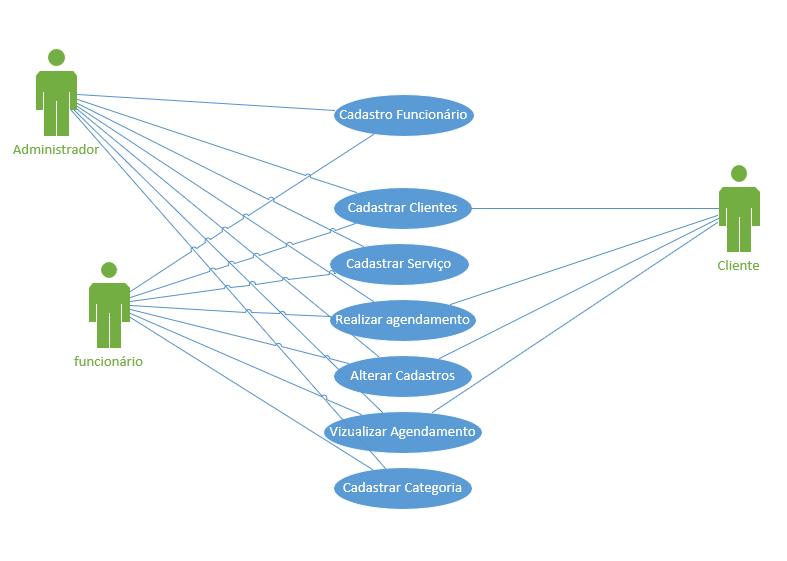
|  |  |  |  |
| --- | --- | --- | --- |
| **Identificador** | RN007 | | |
| **‘Nome** | ACESSO A TELA DE BUSCA | | |
| **Data de criação** | 26/08/2022 | **Autor** | BRUNO |
| **Data da última alteração** | 26/10/2022 | **Autor** | BRUNO |
| **Versão** | 1.0 | **Dependências** | RF006 |
| **Descrição** | SERÁ OBRIGATÓRIO SELECIONAR AO MENOS UM FILTRO PARA A SEPARAÇÃO DOS DADOS. | | |

# **REQUISITOS NÃO FUNCIONAIS**

|  |  |
| --- | --- |
| **ID E TÍTULO** | **DETALHAMENTO** |
| RNF1 – Desempenho: |  |
| RNF2 – Disponibilidade: |  |
| RNF3 – Segurança: |  |
| RNF4 – Interoperabilidade: |  |
| RNF5 – Usabilidade: |  |
| RNF6 – Compatibilidade: |  |
| RNF7 – Acessibilidade: |  |

# **Diagrama e Descrição do Caso de Uso**

3.1 Diagrama



3.2 Descrição

|  |
| --- |
| **Narrativa de caso de uso** |
| **Nome do Caso de Uso:** Realizar agendamento  **Objetivo:** Otimizar e automatizar o agendamento de serviços.  **Ator principal:** Cliente, Funcionário, administrador  **Descrição:** Realizar o agendamento de uma data e horário para um serviço, como corte ou pintura. |
| **CENÁRIO PRINCIPAL**  Na homepage o cliente selecionará a opção “Agendamento”, onde será redirecionado a outra aba para, onde aparecerá a opção “Agendar Serviço”, que o mesmo irá selecionar. Após selecionar, será solicitado o tipo de serviço, o serviço, a data e horário em que deseja. |
| **CENÁRIOS ALTERNATIVOS**  No seu perfil o usuário poderá selecionar para visualizar seus agendamentos, após essa opção surgirá como opção “Novo agendamento”, onde ele poderá realizar um novo agendamento, onde será redirecionado para a aba “Agendar Serviço”, que será solicitada o tipo de serviço, o serviço a ser executada, a data e o horário em que o cliente deseja. |
| **CENÁRIOS DE EXCEÇÃO** |
| **PRÉ-CONDIÇÕES**  Para efetuar um agendamento é necessário que tenha efetuado o cadastro anteriormente e esteja logado. |
| **PÓS-CONDIÇÕES**  Após realizar o agendamento, será gerado um comprovante de agendamento. |

|  |
| --- |
| **Narrativa de caso de uso** |
| **Nome do Caso de Uso:** Alterar Cadastro de Cliente  **Objetivo:** Alteração de dados do cadastro do cliente no sistema.  **Ator principal:** Cliente, administrador, Funcionário  **Descrição:** Alterar os dados do cliente no sistema para que as informações fiquem sempre atualizadas. |
| **CENÁRIO PRINCIPAL**  Após estiver logado, o cliente terá a opção “Perfil” na homepage, a partir dessa opção ele terá a opção “Agendamentos” e “Editar Perfil”, onde o mesmo irá selecionar “Editar Perfil” para que ele possa alterar o cadastro. Onde ele poderá alterar o endereço e telefone e o login (senha e usuário) e e-mail. |
| **CENÁRIOS ALTERNATIVOS** |
| **CENÁRIOS DE EXCEÇÃO** |
| **PRÉ-CONDIÇÕES**  Para alterar o cadastro é necessário que o cliente esteja logado. |
| **PÓS-CONDIÇÕES**  Após realizar a alteração será notificado que houve a alteração. |

|  |
| --- |
| **Narrativa de caso de uso** |
| **Nome do Caso de Uso:** Alterar Cadastro de Funcionário  **Objetivo:** Alteração de dados do cadastro do Funcionário no sistema.  **Ator principal:** Funcionário, administrador  **Descrição:** Alterar os dados do cliente no sistema para que as informações fiquem sempre atualizadas. |
| **CENÁRIO PRINCIPAL**  pós estiver logado, o funcionário terá a opção “Funcionário” na homepage, a partir dessa opção ele poderá verificar os clientes cadastros, e terá a opção para selecionar um cliente. Após selecionar ele terá os dados do cliente e a opção de “Editar Perfil”, onde o mesmo irá selecionar para que ele possa alterar o cadastro. Onde ele poderá alterar o endereço e telefone, nome e login (senha e usuário) e e-mail. |
| **CENÁRIOS ALTERNATIVOS** |
| **CENÁRIOS DE EXCEÇÃO** |
| **PRÉ-CONDIÇÕES**  Para realizar a alteração é necessário que esteja logado em uma conta com nível de funcionário. |
| **PÓS-CONDIÇÕES**  Após realizar a alteração aparecerá a notificação de que foi realizada a alteração e o cliente será notificado. |

|  |
| --- |
| **Narrativa de caso de uso** |
| **Nome do Caso de Uso:** Cadastrar Clientes  **Objetivo:** Realizar o cadastro de clientes que vão na barbearia.  **Ator principal:** Funcionário, administrador  **Descrição:** Realizar o cadastro de clientes para que o acesso as informações fiquem mais fácil e otimizar o atendimento. |
| **CENÁRIO PRINCIPAL**  Na homepage o funcionário terá a opção “Cadastrar”, logo após o mesmo clicar, irá aparecer a opção “Cadastrar Clientes”, onde o mesmo irá selecionar e será solicitado os dados do cliente, como nome, telefone, cpf, e-mail e endereço, será possível criar também o login do cliente. |
| **CENÁRIOS ALTERNATIVOS** |
| **CENÁRIOS DE EXCEÇÃO** |
| **PRÉ-CONDIÇÕES**  É necessário que esteja logado em uma conta como funcionário para que seja possível realizar os cadastros de clientes. |
| **PÓS-CONDIÇÕES**  Será emitido uma notificação que o cliente foi cadastrado. |

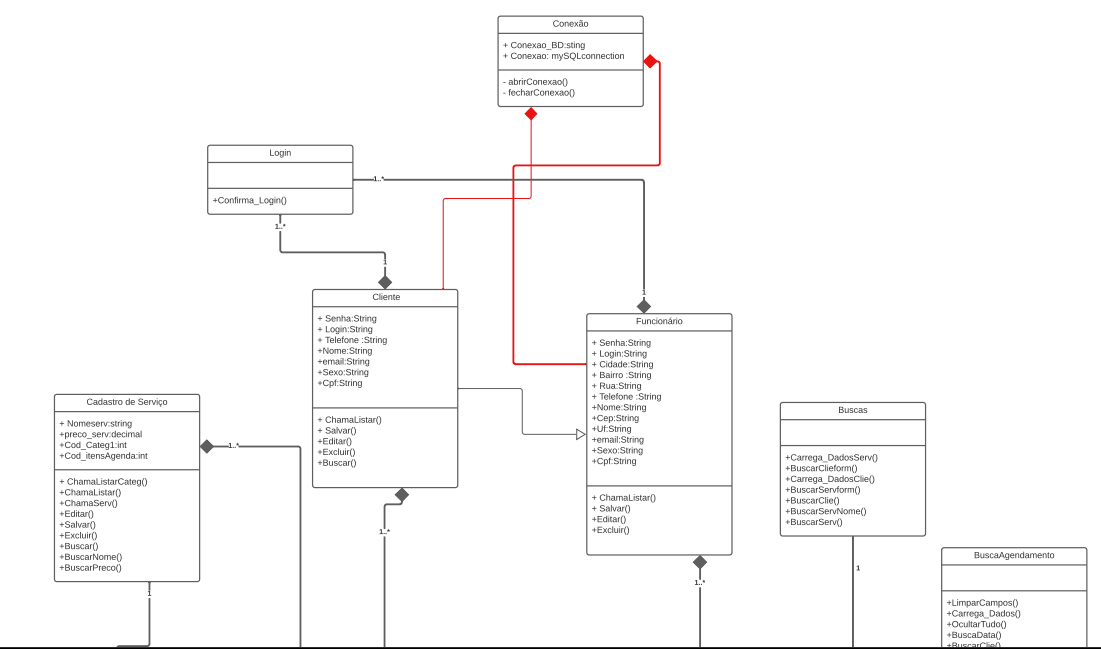
|  |
| --- |
| **Narrativa de caso de uso** |
| **Nome do Caso de Uso:** Cadastrar Funcionário  **Objetivo:** Realizar o cadastro de funcionário.  **Ator principal:** Administrador  **Descrição:** Realizar o cadastro de funcionários para que os mesmos tenham acesso ao sistema com permissões de acesso superior. |
| **CENÁRIO PRINCIPAL**  O administrador terá acesso a seção “Painel” que ficará no cabeçalho da página, onde o mesmo terá a opções de administrador, o mesmo irá selecionar a opção “Cadastrar Funcionário” e logo após será solicitado os dados do funcionário, como: Nome, telefone, endereço, cpf e e-mail. Após isso será solicitado o login e senha do funcionário. |
| **CENÁRIOS ALTERNATIVOS** |
| **CENÁRIOS DE EXCEÇÃO** |
| **PRÉ-CONDIÇÕES**  É necessário que esteja logado em uma conta de administrador. |
| **PÓS-CONDIÇÕES**  Após isso será gerado uma notificação que a conta foi cadastrada. |

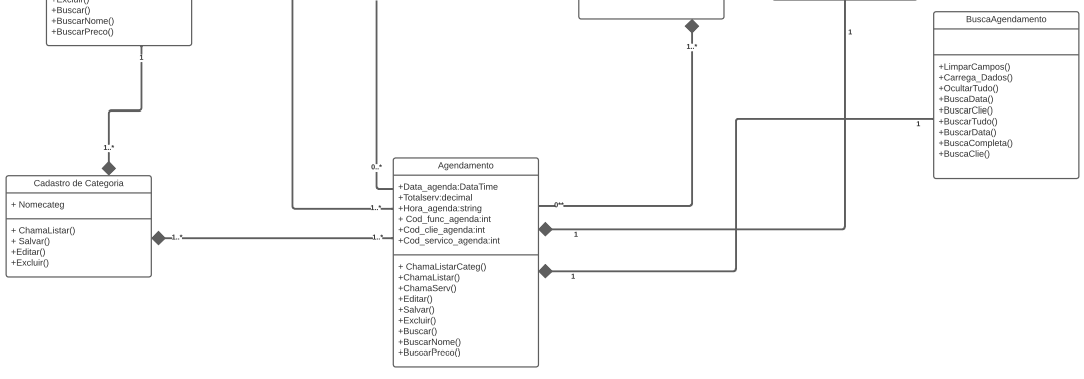
|  |
| --- |
| **Narrativa de caso de uso** |
| **Nome do Caso de Uso:** Criar Categoria  **Objetivo:** Criar categorias de Serviço.  **Ator principal:** Funcionário, Administrador  **Descrição:** Criar uma categoria de atuação de serviço para que esteja bem divido as funções. |
| **CENÁRIO PRINCIPAL**  Tela de criação de categoria sendo inserido o nome da categoria. |
| **CENÁRIOS ALTERNATIVOS**  Depois de criado pode ser direcionado diretamente para a tela de cadastro de serviço. |
| **CENÁRIOS DE EXCEÇÃO**  Se já existir irá dar erro mostrando em uma caixa de texto. |
| **PRÉ-CONDIÇÕES**  Ter acesso ao sistema. |
| **PÓS-CONDIÇÕES**  Estará livre para |

|  |
| --- |
| **Narrativa de caso de uso** |
| **Nome do Caso de Uso:** Cadastrar Serviço  **Objetivo:** Realizar o cadastro dos serviços.  **Ator principal:** Funcionário  **Descrição:** Realizar o cadastro dos serviços para que a barbearia possa seguir com os serviços automatizados, agilizando e organizando o trabalho |
| **CENÁRIO PRINCIPAL**  Será na tela de cadastro de serviço tendo que ser inserido os dados de nome, valor e que categoria ele atende |
| **CENÁRIOS ALTERNATIVOS** |
| **CENÁRIOS DE EXCEÇÃO** |
| **PRÉ-CONDIÇÕES**  Precisa já ter criado uma categoria anteriormente e estar sob domínio do usuário de Funcionário ou administrador |
| **PÓS-CONDIÇÕES**  Gera uma mensagem de cadastro efetivado e com isso já pode ter acesso funcional a tela de agendamento. |

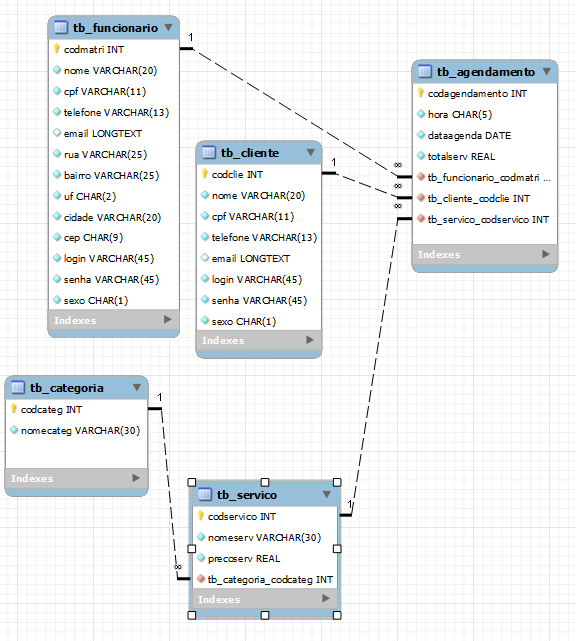
|  |
| --- |
| **Narrativa de caso de uso** |
| **Nome do Caso de Uso:** Visualizar agendamentos  **Objetivo:** poder ter acesso aos agendamentos que já foram feitos, ou que serão feitos  **Ator principal:** Funcionário, Administrador  **Descrição:** poder ter o controle de quais serviços terão que ser feitos em cada dia, semana ou mês |
| **CENÁRIO PRINCIPAL**  Tela de Visualização dos agendamentos, sendo mostrados por dia, semana ou mês |
| **CENÁRIOS ALTERNATIVOS**  Caso não tenha feito o cadastro do cliente naquele momento, poderá ser redirecionado para a tela de cadastro de clientes para aí sim progredir o agendamento. |
| **CENÁRIOS DE EXCEÇÃO**  Se não houver o horário selecionado, irá aparecer uma mensagem mostrando que aquele horário está ocupado, podendo alterar para outro horário. |
| **PRÉ-CONDIÇÕES**  Ter o acesso específico para ver determinadas funções. |
| **PÓS-CONDIÇÕES**  Somente ficar atento aos SMS´s enviados para ficar atento e não perder o seu corte! |

# **Diagrama de Classe**





# **Diagrama de Entidade e Relacionamento (DER)**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ENTIDADE FORTE: tb\_categoria** | | | | |
| **ATRIBUTO** | **CLASSE** | **DOMÍNIO** | **TAMANHO** | **DESCRIÇÃO** |
| codcateg | Identificador | int | X | Código da categoria, gerado automaticamente, não haverá repetições. |
| nome | Obrigatório | varchar |  | Nome da categoria. Ex: Cortes, pinturas, produtos para cabelo. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ENTIDADE FORTE: tb\_funcionario** | | | | |
| **ATRIBUTO** | **CLASSE** | **DOMÍNIO** | **TAMANHO** | **DESCRIÇÃO** |
| codmatri | Identificador | int | x | Matrícula do funcionário, gerado automaticamente, não haverá repetições. |
| Nome | Obrigatório | varchar | 20 | Nome do funcionário. |
| CPF | Obrigatório | Varchar | 11 | CPF do funcionário. |
| Email | Opcional | Varchar | 30 | Email do funcionário. |
| Telefone | Multivalorado | Varchar | 13 | Poderá ser cadastro o telefone fixo, móvel e recado do funcionário. |
| Endereço | Composto | Varchar | 81 | Será cadastro endereço do funcionário, será solicitado, a rua, a cidade, bairro, uf, e cep. |
| LOGIN | Obrigatório | Varchar | 45 | Será salvo junto com o cadastro do cliente, é uma forma de segurança para o sistema |
| SENHA | Obrigatório | Varchar | 45 | Será salvo junto com o cadastro do cliente, é uma forma de segurança para o sistema |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ENTIDADE FRACA: tb\_cliente** | | | | |
| **ATRIBUTO** | **CLASSE** | **DOMÍNIO** | **TAMANHO** | **DESCRIÇÃO** |
| codclie | Identificador | int | x | Código do cliente, gerado automaticamente, não haverá repetições. |
| Nome | Obrigatório | varchar | 20 | Nome do cliente. |
| CPF | Obrigatório | Varchar | 11 | CPF do cliente. |
| Email | Opcional | Varchar | 30 | Email do cliente. |
| Telefone | Multivalorado | Varchar | 13 | Poderá ser cadastro o telefone fixo, móvel e recado do cliente. |
| login | obrigatório | varchar | 45 | Será salvo junto com o cadastro do cliente, é uma forma de segurança para o sistema |
| senha | obrigatório | varchar | 45 | Será salvo junto com o cadastro do cliente, é uma forma de segurança para o sistema |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ENTIDADE FRACA: tb\_servico** | | | | |
| **ATRIBUTO** | **CLASSE** | **DOMÍNIO** | **TAMANHO** | **DESCRIÇÃO** |
| Codserviço | Identificador | int | X | Código do serviço, gerado automaticamente, não haverá repetições. |
| Nomeserv | Obrigatório | varchar | 30 | Nome do serviço, como: Corte Surfista, pintura, luzes. |
| Precoserv | Obrigatório | Real | X | Preço do serviço. |
| Codcateg | Estrangeiro | Int | X | Código da categoria, que surgirá com o relacionamento com a tabela tb\_categoria. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ENTIDADE FRACA: tb\_agendamento** | | | | |
| **ATRIBUTO** | **CLASSE** | **DOMÍNIO** | **TAMANHO** | **DESCRIÇÃO** |
| codagendamento | Identificador | int | X | Código do agedamento, gerado automaticamente, não haverá repetições. |
| Dataagenda | Obrigatório | Data | 8 | Data em que será agendada para realizar o serviço. |
| totalserv | obrigatório | real | x | Será a soma dos serviços que resultara o resultado |
| Tb\_funcionario\_codmatri | Estrangeiro | Int | X | Código do funcionário, chave estrangeiro |
| Tb\_cliente\_codclie | Estrangeiro | int | X | Código do cliente, chave estrangeira |
| Tb\_servico\_codservico | Estrangeiro | int | X | Código de Serviço, Chave estrangeira |
| hora | Obrigatório | Varchar | 5 | Hora em que será agendada para realizar o serviço. |

# **Script SQL**

CREATE DATABASE IF NOT EXISTS `sistemabarbearia` /\*!40100 DEFAULT CHARACTER SET utf8 \*/;

USE `sistemabarbearia`;

-- MySQL dump 10.13 Distrib 8.0.26, for Win64 (x86\_64)

--

-- Host: localhost Database: sistemabarbearia

-- ------------------------------------------------------

-- Server version 5.7.35-log

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!50503 SET NAMES utf8 \*/;

/\*!40103 SET @OLD\_TIME\_ZONE=@@TIME\_ZONE \*/;

/\*!40103 SET TIME\_ZONE='+00:00' \*/;

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

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

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

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

--

-- Table structure for table `tb\_agendamento`

--

DROP TABLE IF EXISTS `tb\_agendamento`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!50503 SET character\_set\_client = utf8mb4 \*/;

CREATE TABLE `tb\_agendamento` (

`codagendamento` int(11) NOT NULL AUTO\_INCREMENT,

`hora` char(5) NOT NULL,

`dataagenda` date NOT NULL,

`totalserv` double NOT NULL,

`tb\_funcionario\_codmatri` int(11) NOT NULL,

`tb\_cliente\_codclie` int(11) NOT NULL,

`tb\_servico\_codservico` int(11) NOT NULL,

PRIMARY KEY (`codagendamento`),

KEY `fk\_tb\_agendamento\_tb\_funcionario1\_idx` (`tb\_funcionario\_codmatri`),

KEY `fk\_tb\_agendamento\_tb\_cliente1\_idx` (`tb\_cliente\_codclie`),

KEY `fk\_tb\_agendamento\_tb\_servico1\_idx` (`tb\_servico\_codservico`),

CONSTRAINT `fk\_tb\_agendamento\_tb\_cliente1` FOREIGN KEY (`tb\_cliente\_codclie`) REFERENCES `tb\_cliente` (`codclie`) ON DELETE NO ACTION ON UPDATE NO ACTION,

CONSTRAINT `fk\_tb\_agendamento\_tb\_funcionario1` FOREIGN KEY (`tb\_funcionario\_codmatri`) REFERENCES `tb\_funcionario` (`codmatri`) ON DELETE NO ACTION ON UPDATE NO ACTION,

CONSTRAINT `fk\_tb\_agendamento\_tb\_servico1` FOREIGN KEY (`tb\_servico\_codservico`) REFERENCES `tb\_servico` (`codservico`) ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Dumping data for table `tb\_agendamento`

--

LOCK TABLES `tb\_agendamento` WRITE;

/\*!40000 ALTER TABLE `tb\_agendamento` DISABLE KEYS \*/;

/\*!40000 ALTER TABLE `tb\_agendamento` ENABLE KEYS \*/;

UNLOCK TABLES;

--

-- Table structure for table `tb\_categoria`

--

DROP TABLE IF EXISTS `tb\_categoria`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!50503 SET character\_set\_client = utf8mb4 \*/;

CREATE TABLE `tb\_categoria` (

`codcateg` int(11) NOT NULL AUTO\_INCREMENT,

`nomecateg` varchar(30) NOT NULL,

PRIMARY KEY (`codcateg`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Dumping data for table `tb\_categoria`

--

LOCK TABLES `tb\_categoria` WRITE;

/\*!40000 ALTER TABLE `tb\_categoria` DISABLE KEYS \*/;

/\*!40000 ALTER TABLE `tb\_categoria` ENABLE KEYS \*/;

UNLOCK TABLES;

--

-- Table structure for table `tb\_cliente`

--

DROP TABLE IF EXISTS `tb\_cliente`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!50503 SET character\_set\_client = utf8mb4 \*/;

CREATE TABLE `tb\_cliente` (

`codclie` int(11) NOT NULL AUTO\_INCREMENT,

`nome` varchar(20) NOT NULL,

`cpf` varchar(11) NOT NULL,

`telefone` varchar(13) NOT NULL,

`email` longtext,

`login` varchar(45) NOT NULL,

`senha` varchar(45) NOT NULL,

`sexo` char(1) NOT NULL,

PRIMARY KEY (`codclie`),

UNIQUE KEY `cpf\_UNIQUE` (`cpf`)

) ENGINE=InnoDB AUTO\_INCREMENT=2 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Dumping data for table `tb\_cliente`

--

LOCK TABLES `tb\_cliente` WRITE;

/\*!40000 ALTER TABLE `tb\_cliente` DISABLE KEYS \*/;

INSERT INTO `tb\_cliente` VALUES (1,'Felipy','12345678901','25896314','shgdv','adm','adm','F');

/\*!40000 ALTER TABLE `tb\_cliente` ENABLE KEYS \*/;

UNLOCK TABLES;

--

-- Table structure for table `tb\_funcionario`

--

DROP TABLE IF EXISTS `tb\_funcionario`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!50503 SET character\_set\_client = utf8mb4 \*/;

CREATE TABLE `tb\_funcionario` (

`codmatri` int(11) NOT NULL AUTO\_INCREMENT,

`nome` varchar(20) NOT NULL,

`cpf` varchar(11) NOT NULL,

`telefone` varchar(13) NOT NULL,

`email` longtext,

`rua` varchar(25) NOT NULL,

`bairro` varchar(25) NOT NULL,

`uf` char(2) NOT NULL,

`cidade` varchar(20) NOT NULL,

`cep` char(9) NOT NULL,

`login` varchar(45) NOT NULL,

`senha` varchar(45) NOT NULL,

`sexo` char(1) NOT NULL,

PRIMARY KEY (`codmatri`),

UNIQUE KEY `cpf\_UNIQUE` (`cpf`)

) ENGINE=InnoDB AUTO\_INCREMENT=2 DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Dumping data for table `tb\_funcionario`

--

LOCK TABLES `tb\_funcionario` WRITE;

/\*!40000 ALTER TABLE `tb\_funcionario` DISABLE KEYS \*/;

INSERT INTO `tb\_funcionario` VALUES (1,'hsagd','12345678901','2646546','hdg','asjhgd','ajhdsf','es','wre','25896314','adm','adm','f');

/\*!40000 ALTER TABLE `tb\_funcionario` ENABLE KEYS \*/;

UNLOCK TABLES;

--

-- Table structure for table `tb\_servico`

--

DROP TABLE IF EXISTS `tb\_servico`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!50503 SET character\_set\_client = utf8mb4 \*/;

CREATE TABLE `tb\_servico` (

`codservico` int(11) NOT NULL AUTO\_INCREMENT,

`nomeserv` varchar(30) NOT NULL,

`precoserv` double NOT NULL,

`tb\_categoria\_codcateg` int(11) NOT NULL,

PRIMARY KEY (`codservico`),

KEY `fk\_tb\_servico\_tb\_categoria\_idx` (`tb\_categoria\_codcateg`),

CONSTRAINT `fk\_tb\_servico\_tb\_categoria` FOREIGN KEY (`tb\_categoria\_codcateg`) REFERENCES `tb\_categoria` (`codcateg`) ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Dumping data for table `tb\_servico`

--

LOCK TABLES `tb\_servico` WRITE;

/\*!40000 ALTER TABLE `tb\_servico` DISABLE KEYS \*/;

/\*!40000 ALTER TABLE `tb\_servico` ENABLE KEYS \*/;

UNLOCK TABLES;

--

-- Dumping events for database 'sistemabarbearia'

--

--

-- Dumping routines for database 'sistemabarbearia'

--

/\*!40103 SET TIME\_ZONE=@OLD\_TIME\_ZONE \*/;

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

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

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

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

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

-- Dump completed on 2022-10-27 11:36:16

# **Casos de Teste**

# **6.1 Especificação dos Casos de Teste**

## **6.1.1 Caso de Teste**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ESPECIFICAÇÃO DE CASO DE TESTE** | | | | |
| **Título** | | VALIDAÇÃO DO CADASTRO DE CLIENTES | | |
| **Caso de teste** | | CT-001 | | |
| **Tipo** | | FUNCIONAL | | |
| **Requisito Funcional** | | RF001 | | |
| **Propósito** | | <<  OBJETIVO GERAL DO TESTE: VALIDAR A INSERÇÃO DOS DADOS DO CLIENTE NO BANCO  >> | | |
| **Descrição geral**  <<  Será testado e validado os dados que devem ser armazenados do cliente  >> | | | | |
| **Insumos para o caso de teste**  <<  Antes do teste o MySQL deve ser iniciado  >> | | | | |
| **Roteiro para a realização do teste**  <<  Entrar no diretório S:\#Turma 76 - Tec em Informática\UC-16\Vinícius - Felipy - Bruno\Versões\Versão WEB\SistemaBarbearia  Duplo clique no arquivo SistemaBarbearia.Sln  No menu, clicar em “Cadastrar” logo após em “Cadastrar Clientes”  Em cadastro de clientes, deverá apertar o botão “Novo” para cadastrar um novo funcionário.  >> | | | | |
| **Cenários de teste** | | | | |
| Objetivo específico | | | Especificação das entradas ou ações | Saídas esperadas |
| 1 | Validar os campos nome, telefone e cpf | | Os campo nome, cpf e telefone DEVEM estar vazios | O sistema deverá emitir uma mensagem de erro sobre os campos vazios. Depois deverá limpar todos os campos posicionar o cursor no campo nome |
| 2 | Verificar se após as mensagens dos erros os botões retornam para a posição inicial | | Ao clicar em novo ou em editar os botões tomam uma nova posição, deve ser verificado se esses botões retomam as posições iniciais. | Botões retornarem as posições iniciais. |

## **6.1.2 Caso de Teste**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ESPECIFICAÇÃO DE CASO DE TESTE** | | | | |
| **Título** | | Validação da tela de Busca | | |
| **Caso de teste** | | CT-002 | | |
| **Tipo** | | FUNCIONAL | | |
| **Requisito Funcional** | | RF007 | | |
| **Propósito** | | <<  OBJETIVO GERAL DO TESTE: VALIDAR A BUSCA DE DADOS CONFORME O FILTRO SELECIONADO PELO USUÁRIO  >> | | |
| **Descrição geral**  <<  Será testado e validados os resultados das buscas.  >> | | | | |
| **Insumos para o caso de teste**  <<  É necessário iniciar o banco de dados antes de realizar o teste.  >> | | | | |
| **Roteiro para a realização do teste**  <<  Entrar no diretório S:\#Turma 76 - Tec em Informática\UC-16\Vinícius - Felipy - Bruno\Versões\Versão WEB\SistemaBarbearia  Duplo clique no arquivo SistemaBarbearia.Sln  No menu, clicar em “Buscar” logo após selecione o tipo de busca que deseja testar  >> | | | | |
| **Cenários de teste** | | | | |
| Objetivo específico | | | Especificação das entradas ou ações | Saídas esperadas |
| 1 | Verificar se os formulários estão recebendo os valores conforme o a seleção de busca escolhida no menu. | | Ao selecionar um tipo de busca no menu, o formulário deverá carregas os campos conforme a opção. | Sistema deverá trazer os campos de busca conforme o filtro selecionado. |
| 2 | Verificar se os dados do resultado da busca estão corretos. | | Verificar no banco se os dados apresentados no resultado são os mesmos cadastrados. | Os dados devem ser os mesmo para o banco e para o resultado. |

## **6.1.3 Caso de Teste**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ESPECIFICAÇÃO DE CASO DE TESTE** | | | | |
| **Título** | | Validação cadastro do funcionário. | | |
| **Caso de teste** | | CT-003 | | |
| **Tipo** | | FUNCIONAL | | |
| **Requisito Funcional** | | RF-006 | | |
| **Propósito** | | <<  INSERÇÃO DOS DADOS DO FUNCIONÁRIO NO BANCO.  >> | | |
| **Descrição geral**  <<  SERÁ TESTADO E VALIDADO OS DADOS QUE DEVEM SER OBRIGATÓRIO NO CADASTRO DO FUNCIONARIO.  >> | | | | |
| **Insumos para o caso de teste**  <<  ANTES DO TESTE O MYSQL DEVE SER ATIVADO.  >> | | | | |
| **Roteiro para a realização do teste**  <<  Entrar no diretório S:\#Turma 76 - Tec em Informática\UC-16\Vinícius - Felipy - Bruno\Versões\Versão WEB\SistemaBarbearia  Duplo clique no arquivo SistemaBarbearia.Sln  Iniciar o programa.  No menu, clicar em “Cadastrar” logo após em “cadastro funcionário”  Preencher os campos que são obrigatórios para o cadastro, testar todas as formas  Validar o programa.  >> | | | | |
| **Cenários de teste** | | | | |
| Objetivo específico | | | Especificação das entradas ou ações | Saídas esperadas |
| 1 | Validar os campos nome, telefone e cpf | | Os campos nome, telefone e cpf devem estar vazios. | Para cada campo deixado vazio, uma mensagem de erro surgira, limpando todos os campos e colocando o cursor novamente no campo nome |
| 2 | Validar se o cadastro do funcionário está sendo salvo no banco | | Preencher todos os campos e verificar se foi salvo no banco. | Uma mensagem que foi salvo com sucesso. |

## **6.1.4 Caso de Teste**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ESPECIFICAÇÃO DE CASO DE TESTE** | | | | |
| **Título** | | Validação do agendamento | | |
| **Caso de teste** | | CT-004 | | |
| **Tipo** | | FUNCIONAL | | |
| **Requisito Funcional** | | RF-002 | | |
| **Propósito** | | <<  Testar data, hora e funcionário diferentes para validação do agendamento, se está salvando no banco e verificar se está tendo agendamento duplicado.  >> | | |
| **Descrição geral**  <<  SERÁ TESTADO E VALIDADO OS DADOS QUE DEVEM SER OBRIGATÓRIO NO AGENDAMENTO.  >> | | | | |
| **Insumos para o caso de teste**  <<  ANTES DO TESTE O MYSQL DEVE SER ATIVADO.  >> | | | | |
| **Roteiro para a realização do teste**  <<  Entrar no diretório S:\#Turma 76 - Tec em Informática\UC-16\Vinícius - Felipy - Bruno\Versões\Versão WEB\SistemaBarbearia  Duplo clique no arquivo SistemaBarbearia.Sln  Iniciar o programa.  No menu, clicar em “agendamento” logo após em “agendar”  Preencher os campos que são obrigatórios para o agendamento, testar todas as formas para validar o agendamento,  E verificar se não está duplicando agendamentos.  Validar o programa.  >> | | | | |
| **Cenários de teste** | | | | |
| Objetivo específico | | | Especificação das entradas ou ações | Saídas esperadas |
| 1 | Validar os campos data, hora e fucionário. | | Os campos data, hora e funcionario devem estar vazios/nulos. | Para cada campo deixado vazio, uma mensagem de erro surgira, limpando todos os campos e colocando o cursor novamente no campo nome |
| 2 | Validar o agendamento, olhando no banco e verificando se existe agendamentos duplicados. | | Preencher todos os campos e verificar se foi salvo no banco. | Uma mensagem que foi salvo com sucesso. |

**6.1.5 Caso de Teste**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ESPECIFICAÇÃO DE CASO DE TESTE** | | | | |
| **Título** | | VALIDAÇÃO DO LOGIN | | |
| **Caso de teste** | | CT-005 | | |
| **Tipo** | | FUNCIONAL | | |
| **Requisito Funcional** | | RF-003 | | |
| **Propósito** | | <<  OBJETIVO GERAL DO TESTE: VALIDAR OS DADOS CRIADOS NO BANCO DE DADOS PARA O ACESSO AO SISTEMA.  >> | | |
| **Descrição geral**  <<  Será testado e validado os dados que foram adicionados no banco para o acesso ao sistema.  >> | | | | |
| **Insumos para o caso de teste**  <<  Antes do teste o MySQL deve ser iniciado  >> | | | | |
| **Roteiro para a realização do teste**  <<  Entrar no diretório S:\#Turma 76 - Tec em Informática\UC-16\Vinícius - Felipy - Bruno\Versões\Versão WEB\SistemaBarbearia  Duplo clique no arquivo SistemaBarbearia.Sln  No menu, clicar em qualquer campo, ele irá mostrar a tela de login para ter o acesso no sistema, depois do acesso cadastrado anteriormente poderá fazer o que está disponibilizado  Caso não tenha cadastro ele irá te redirecionar para a tela de cadastro que deseja.  >> | | | | |
| **Cenários de teste** | | | | |
| Objetivo específico | | | Especificação das entradas ou ações | Saídas esperadas |
| 1 | Validar os campos no banco de dados | | Os campos de login e senha devem estar vazios. | Para cada campo deixado vazio, uma mensagem de erro surgira, limpando todos os campos e colocando o cursor novamente no campo nome |

## **6.1.6 Caso de Teste**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ESPECIFICAÇÃO DE CASO DE TESTE** | | | | |
| **Título** | | VALIDAÇÃO CADASTRO DE SERVIÇOS | | |
| **Caso de teste** | | CT-006 | | |
| **Tipo** | | FUNCIONAL | | |
| **Requisito Funcional** | | RF-004 | | |
| **Propósito** | | <<  OBJETIVO GERAL DO TESTE: VALIDAR E INSERIR OS DADOS CADASTRADOS NO BANCO DE DADO.  >> | | |
| **Descrição geral**  <<  Será testado e validado os dados que foram adicionados no banco para o cadastro de serviços.  >> | | | | |
| **Insumos para o caso de teste**  <<  Antes do teste o MySQL deve ser iniciado  >> | | | | |
| **Roteiro para a realização do teste**  <<  Entrar no diretório S:\#Turma 76 - Tec em Informática\UC-16\Vinícius - Felipy - Bruno\Versões\Versão WEB\SistemaBarbearia  Duplo clique no arquivo SistemaBarbearia.Sln  Ao iniciar a página inicial, clique em “Logar” tela de login para ter o acesso no sistema, digitar o login adm e senha 123456. Após isso, no menu selecione “CADASTRO” depois “CADASTRAR SERVIÇOS” >> | | | | |
| **Cenários de teste** | | | | |
| Objetivo específico | | | Especificação das entradas ou ações | Saídas esperadas |
| 1 | Validar os campos no banco de dados | | Os campos de nome e categoria devem estar vazios | Para cada campo deixado vazio, uma mensagem de erro surgira, limpando todos os campos e colocando o cursor novamente no campo nome |
| 2 | Validar se os dados não tem repetição | | Não poderá cadastrar um serviço duplicado. | Caso o serviço já esteja cadastrado aparecerá uma mensagem de erro informando. |

## **6.1.7 Caso de Teste**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ESPECIFICAÇÃO DE CASO DE TESTE** | | | | |
| **Título** | | VALIDAÇÃO CADASTRO DE CATEGORIA | | |
| **Caso de teste** | | CT-007 | | |
| **Tipo** | | FUNCIONAL | | |
| **Requisito Funcional** | | RF-005 | | |
| **Propósito** | | <<  OBJETIVO GERAL DO TESTE: VALIDAR E INSERIR OS DADOS CADASTRADOS NO BANCO DE DADO.  >> | | |
| **Descrição geral**  <<  Será testado e validado os dados que foram adicionados no banco para o cadastro de serviços.  >> | | | | |
| **Insumos para o caso de teste**  <<  Antes do teste o MySQL deve ser iniciado  >> | | | | |
| **Roteiro para a realização do teste**  <<  Entrar no diretório S:\#Turma 76 - Tec em Informática\UC-16\Vinícius - Felipy - Bruno\Versões\Versão WEB\SistemaBarbearia  Duplo clique no arquivo SistemaBarbearia.Sln  Ao iniciar a página inicial, clique em “Logar” tela de login para ter o acesso no sistema, digitar o login adm e senha 123456. Após isso, no menu selecione “CADASTRO” depois “CADASTRAR CATEGORIA” >> | | | | |
| **Cenários de teste** | | | | |
| Objetivo específico | | | Especificação das entradas ou ações | Saídas esperadas |
| 1 | Validar os campos no banco de dados | | O campo de nome deve estar vazio. | Se o campo for deixado vazio, uma mensagem de erro surgira, limpando todos os campos e colocando o cursor novamente no campo nome |
| 2 | Validar se os dados não têm repetição | | Não poderá cadastrar uma categoria duplicada. | Caso a categoria já esteja cadastrada aparecerá uma mensagem de erro informando. |

# **6. 2 Análise de Risco**

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de Teste | Impacto | Probabilidades | Risco |
| CT-001 | 9 | 2 | 18 |
| CT-002 | 5 | 2 | 10 |
| CT-003 | 9 | 2 | 18 |
| CT-004 | 9 | 2 | 18 |
| CT-005 | 9 | 2 | 18 |
| CT-006 | 5 | 2 | 10 |
| CT-007 | 5 | 5 | 25 |

Nível de Impacto e probabilidade:

2- Baixo

5-Médio

9-Alto

# **Briefing do Site (Descrição/Resumo; Wireframe; Mapa do Site)**

# **Responsabilidades de Cada Componente**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOME** | **DESCRIÇÃO DA AÇÃO**  **RESPONSABILIDADE** | **DATA ENTREGA** | | **STATUS\*** | **JUSTIFICATIVA PARA:**  **NÃO CUMPRIDO OU EM PARTE** |
| **PREVISTO** | **REALIZADO** |
| Grupo | DESEN. REQUISITOS FUNCIONAIS, E DESEN REGRAS DE NEGOCIOS | 31/08 | 27/08 | CUMPRIDO |  |
| FELIPY | DESEN. DER/ SQL – E DESEN ENTIDADES | 31/08 | 29/08 | CUMPRIDO |  |
| VINICIUS | DESEN. DER – E DESEN. ENTIDADES, REQUISITOS FUNCIONAIS. | 31/08 | 29/08 | CUMPRIDO | - |
| Bruno | montagem da funcionalidade do Model e DAO do C# | 16/10 | 16/10 | CUMPRIDO |  |
| Bruno | Verificação e alteração da documentação | 23/10 | 23/10 | CUMPRIDO |  |
| Vinicius | Montagem do View e funcionalidades do Model e DAO do C# | 16/10 | 16/10 | CUMPRIDO |  |
| Felipy | Montagem da funcionalidade do Model e DAO do C# | 16/10 | 16/10 | CUMPRIDO |  |

**\*STATUS: CUMPRINDO, NÃO CUMPRIDO, EM PARTE**