

Trabalho GCC 115 - Engenharia de Software I

Documento de Design

Barbearia Agendamento Digital – Razor

**Alunos:** Álvaro dos Reis Cozadi – 201211012

Danniel Lucas Cardoso – 201510379

Felipe Rezende Lima – 201421171

Marcos Silveira Bello – 201510870

Otavio Andrade - 201510866

Sumário

1. Tecnologias Utilizadas

2. Diagrama Caso de Uso

3. Banco de Dados

4. Protótipos de Telas

**1. Tecnologias utilizadas**

Principais tecnologias para o Desenvolvimento.

Utilizamos tecnologias 100% compatíveis com os mais variados navegadores Web (browsers), para que nossas soluções tenham a maior abrangência possível, a determinação das tecnologias aplicadas em cada projeto dependeram de suas características técnicas e especificações.

* HTML

Do inglês Hyper Text Markup Language, significa Linguagem de Marcação de Hipertexto, ou seja, HTML é uma coleção de estilos que define os vários componentes de um documento Internet. Utilizado em: formatação, documentos multimídia, formulários, hiperlinks clicáveis, imagens gráficas, etc.

* PHP

Linguagem para programar scripts que são processados no servidor, que se incrustam dentro do código HTML. Esta linguagem é gratuita e de multiplataforma, ou seja, roda em servidores de hospedagem com Linux ou Windows.

* CSS

A utilização de CSS (Cascading Style Sheets) é recomendada, pois permite separar estilo visual de conteúdo. Tal fator é fundamental quando se faz o design e estruturação de um website. Quando CSS é utilizado de maneira correta, a manutenção do website torna-se mais simples. Além disso, existem outras vantagens como facilidade de criar estilos visuais múltiplos por exemplo podendo-se oferecer o estilo otimizado para impressão.

* Adobe Flash

Formato de arquivo multimídia de grande apelo visual, muito difundido atualmente na Internet. Animações com efeitos sonoros e interatividade podem ser criadas com esse recurso, tornando as páginas muito mais atraentes e agradáveis.

* JavaScript

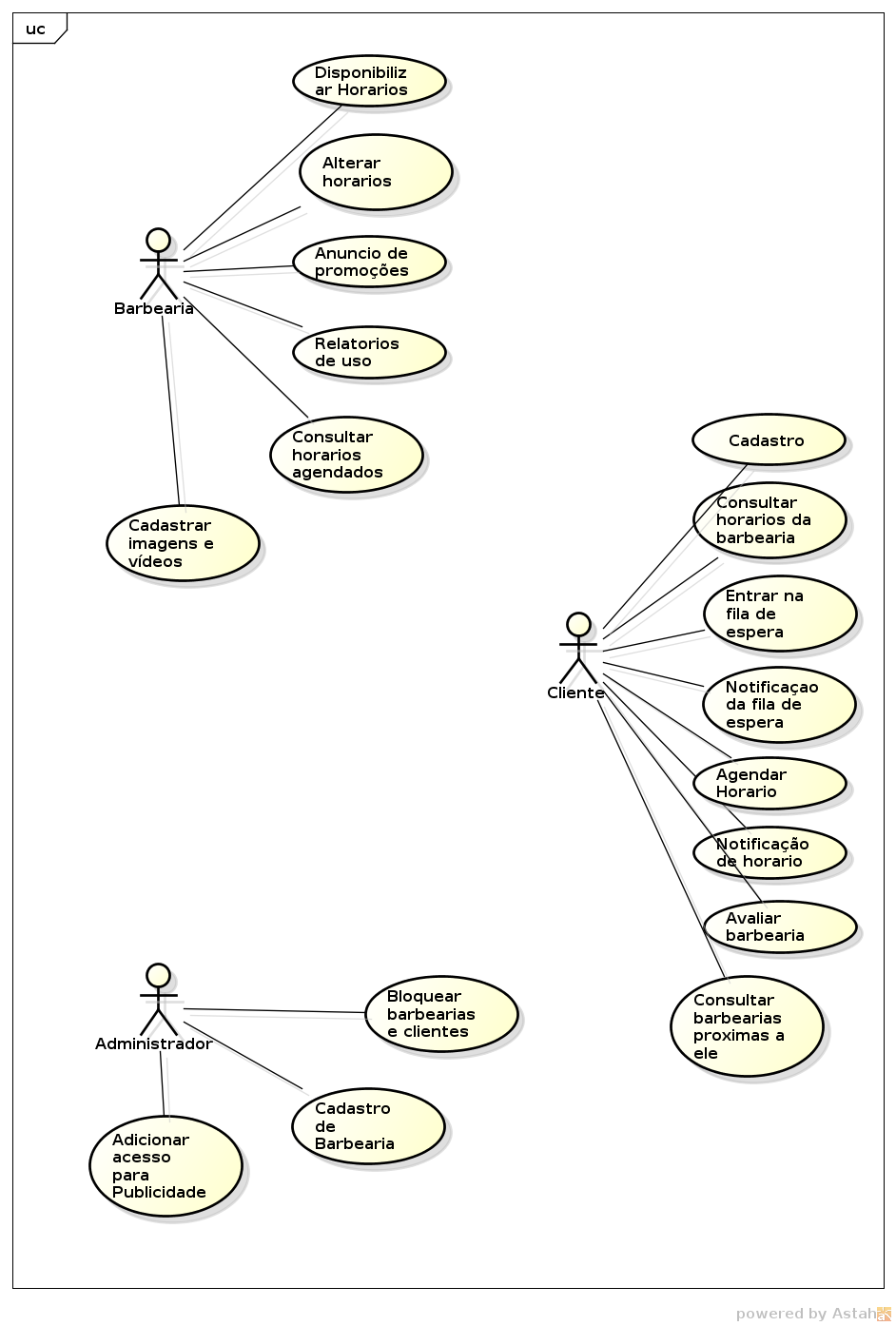
Através do JavaScript você tem muitas possibilidades de melhorar páginas HTML com elementos interessantes. Sendo capaz, por exemplo, de responder muito facilmente a eventos iniciados pelo usuário. Alguns dos efeitos que agora podemos fazer com JavaScript, há algum tempo atrás só eram possíveis com CGI. Deste modo, podemos desenvolver páginas muito sofisticadas com a ajuda do JavaScript.

* MySQL Workbench

Ferramenta gráfica para modelagem de dados, integrando criação e designer que pretende ser uma evolução do já famoso DBDesigner4. A ferramenta possibilita trabalhar diretamente com objetos schema, além de fazer a separação do modelo lógico do catálogo de banco de dados.

* No sistema Razor não será utilizado frameworks para o desenvolvimento.

**2. Diagrama Caso de Uso**

Diagrama caso de uso Razor – agendamento de barbearia

**3. Banco de Dados**

* Arquitetura Banco de dados utilizando MySQL Workbench

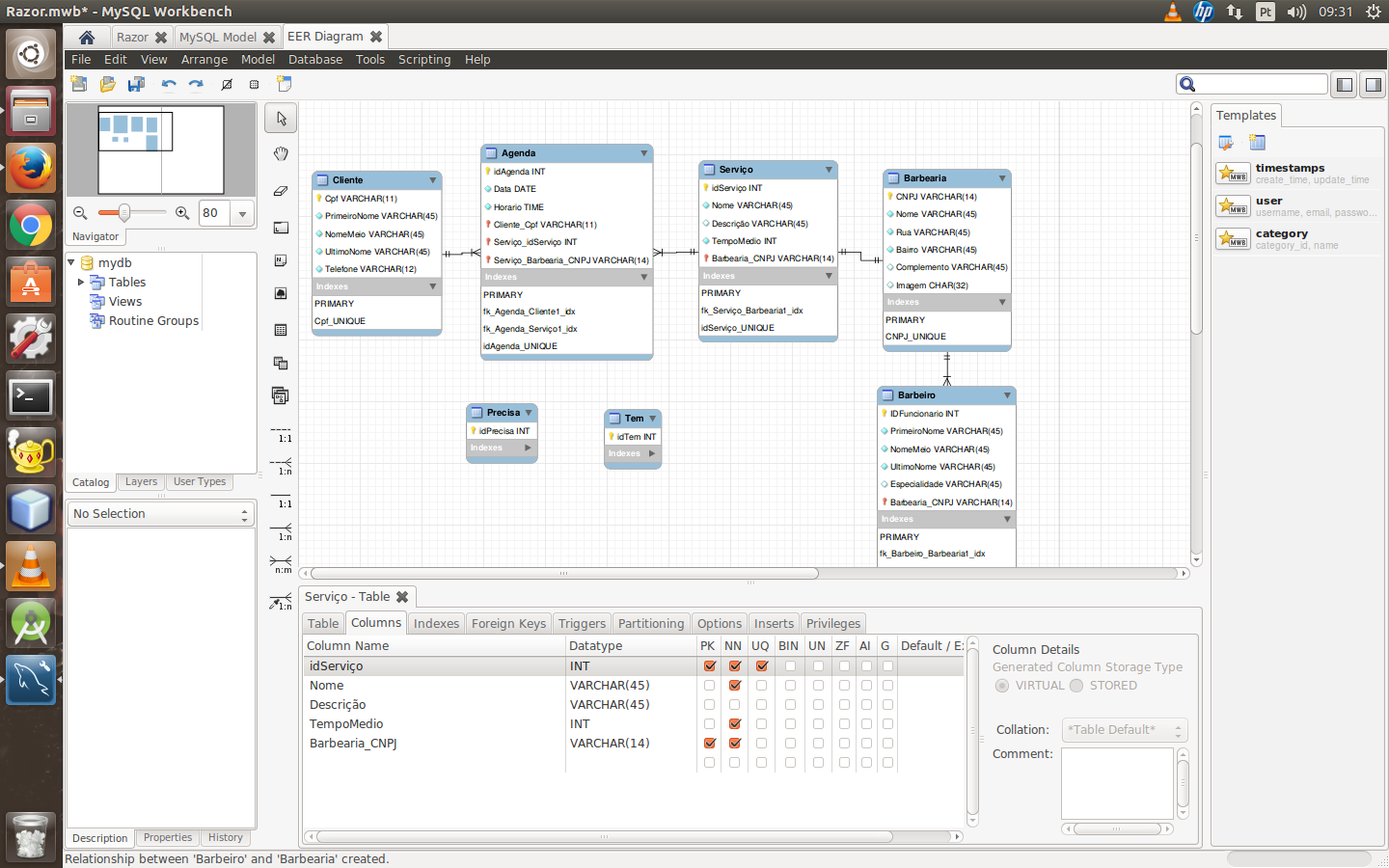


Imagem 1 – Diagrama Banco de Dados

* Código SQL

-- MySQL Script generated by MySQL Workbench

-- Ter 21 Fev 2017 09:37:52 BRT

-- Model: New Model Version: 1.0

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='TRADITIONAL,ALLOW\_INVALID\_DATES';

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

-- Schema mydb

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

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

-- Schema mydb

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

CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;

USE `mydb` ;

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

-- Table `mydb`.`Cliente`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Cliente` (

`Cpf` VARCHAR(11) NOT NULL,

`PrimeiroNome` VARCHAR(45) NOT NULL,

`NomeMeio` VARCHAR(45) NOT NULL,

`UltimoNome` VARCHAR(45) NOT NULL,

`Telefone` VARCHAR(12) NOT NULL,

PRIMARY KEY (`Cpf`),

UNIQUE INDEX `Cpf\_UNIQUE` (`Cpf` ASC))

ENGINE = InnoDB;

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

-- Table `mydb`.`Barbearia`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Barbearia` (

`CNPJ` VARCHAR(14) NOT NULL,

`Nome` VARCHAR(45) NOT NULL,

`Rua` VARCHAR(45) NOT NULL,

`Bairro` VARCHAR(45) NOT NULL,

`Complemento` VARCHAR(45) NULL,

`Imagem` CHAR(32) NULL,

PRIMARY KEY (`CNPJ`),

UNIQUE INDEX `CNPJ\_UNIQUE` (`CNPJ` ASC))

ENGINE = InnoDB;

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

-- Table `mydb`.`ServiÃ§o`

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

CREATE TABLE IF NOT EXISTS `mydb`.`ServiÃ§o` (

`idServiÃ§o` INT NOT NULL,

`Nome` VARCHAR(45) NOT NULL,

`DescriÃ§Ã£o` VARCHAR(45) NULL,

`TempoMedio` INT NOT NULL,

`Barbearia\_CNPJ` VARCHAR(14) NOT NULL,

PRIMARY KEY (`idServiÃ§o`, `Barbearia\_CNPJ`),

INDEX `fk\_ServiÃ§o\_Barbearia1\_idx` (`Barbearia\_CNPJ` ASC),

UNIQUE INDEX `idServiÃ§o\_UNIQUE` (`idServiÃ§o` ASC),

CONSTRAINT `fk\_ServiÃ§o\_Barbearia1`

FOREIGN KEY (`Barbearia\_CNPJ`)

REFERENCES `mydb`.`Barbearia` (`CNPJ`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Agenda`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Agenda` (

`idAgenda` INT NOT NULL,

`Data` DATE NOT NULL,

`Horario` TIME NOT NULL,

`Cliente\_Cpf` VARCHAR(11) NOT NULL,

`ServiÃ§o\_idServiÃ§o` INT NOT NULL,

`ServiÃ§o\_Barbearia\_CNPJ` VARCHAR(14) NOT NULL,

PRIMARY KEY (`idAgenda`, `Cliente\_Cpf`, `ServiÃ§o\_idServiÃ§o`, `ServiÃ§o\_Barbearia\_CNPJ`),

INDEX `fk\_Agenda\_Cliente1\_idx` (`Cliente\_Cpf` ASC),

INDEX `fk\_Agenda\_ServiÃ§o1\_idx` (`ServiÃ§o\_idServiÃ§o` ASC, `ServiÃ§o\_Barbearia\_CNPJ` ASC),

UNIQUE INDEX `idAgenda\_UNIQUE` (`idAgenda` ASC),

CONSTRAINT `fk\_Agenda\_Cliente1`

FOREIGN KEY (`Cliente\_Cpf`)

REFERENCES `mydb`.`Cliente` (`Cpf`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Agenda\_ServiÃ§o1`

FOREIGN KEY (`ServiÃ§o\_idServiÃ§o` , `ServiÃ§o\_Barbearia\_CNPJ`)

REFERENCES `mydb`.`ServiÃ§o` (`idServiÃ§o` , `Barbearia\_CNPJ`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

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

-- Table `mydb`.`Barbeiro`

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

CREATE TABLE IF NOT EXISTS `mydb`.`Barbeiro` (

`IDFuncionario` INT NOT NULL,

`PrimeiroNome` VARCHAR(45) NOT NULL,

`NomeMeio` VARCHAR(45) NOT NULL,

`UltimoNome` VARCHAR(45) NOT NULL,

`Especialidade` VARCHAR(45) NULL,

`Barbearia\_CNPJ` VARCHAR(14) NOT NULL,

PRIMARY KEY (`IDFuncionario`, `Barbearia\_CNPJ`),

INDEX `fk\_Barbeiro\_Barbearia1\_idx` (`Barbearia\_CNPJ` ASC),

UNIQUE INDEX `IDFuncionario\_UNIQUE` (`IDFuncionario` ASC),

CONSTRAINT `fk\_Barbeiro\_Barbearia1`

FOREIGN KEY (`Barbearia\_CNPJ`)

REFERENCES `mydb`.`Barbearia` (`CNPJ`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

**4. Protótipos de Telas**

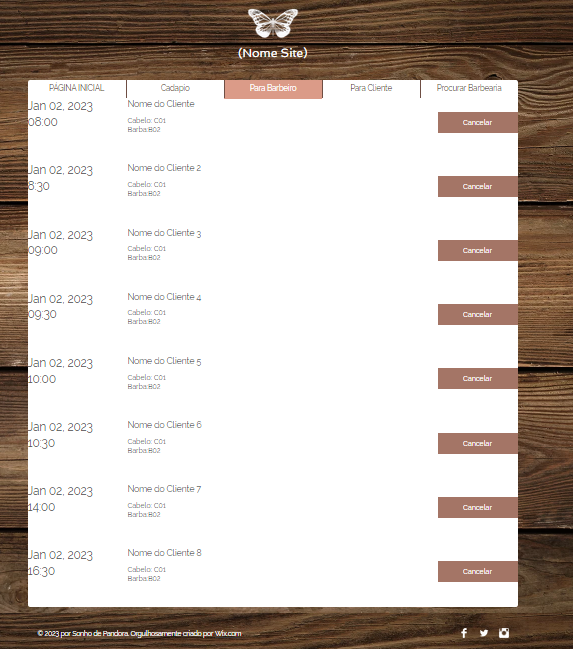


Imagem 2 - Agenda barbeiro



Imagem 3 - Agenda usuário

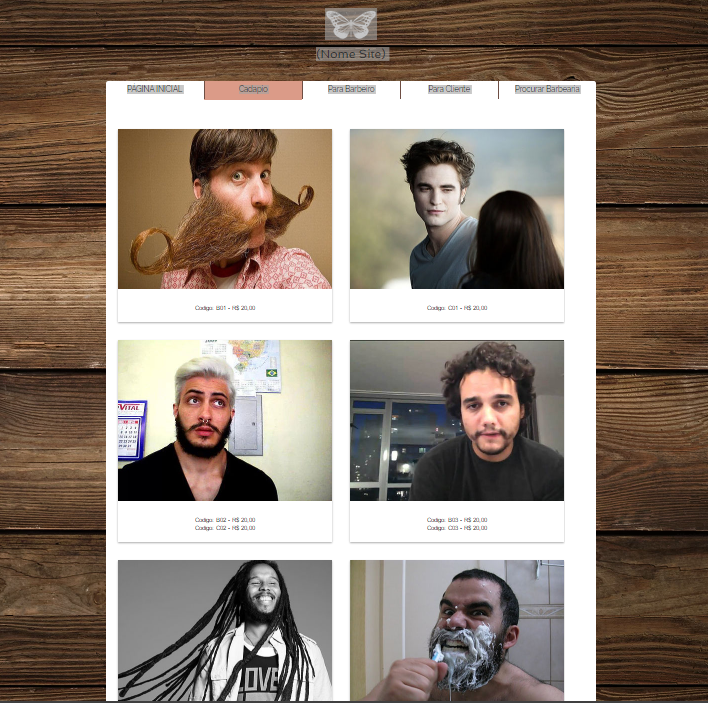


Imagem 4 - Cardápio



Imagem 5 – Procurar barbearia

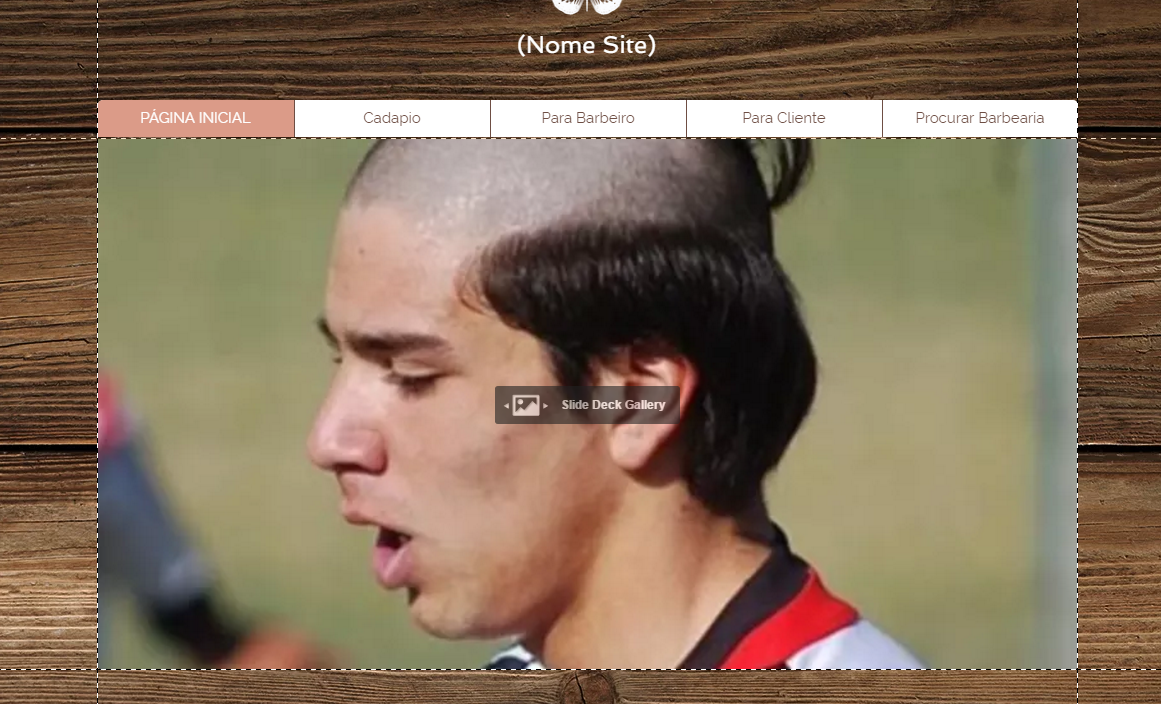


Imagem 6 – Tela principal