Software para gerenciamento de salão de beleza - 7 Beauty

Willow Guimarães Rocha da Silva

Centro Universitário Ritter dos Reis – Laureate International Universities

CEP 90840-440 – Porto Alegre – RS – Brasil

willow\_grs@hotmail.com

**Abstract.** This article intend to document the project of software called “7 Beauty”, that will be developed using PHP language, and initially intend to beauty salons, but in future can be adapted to suit different niches, in the products and services branch. Its goal is to help micro entrepreneurs to manage better their business, keeping track through automation and better visualization of items like customer services, stock and cash flow.

**Resumo.** Este artigo tem a intenção de documentar o projeto do software “7 Beauty” que será desenvolvido usando linguagem PHP, e destina-se inicialmente a salões de beleza, mas que futuramente poderá ser adaptado para atender outros nichos, na área de produtos e serviços. Tem como objetivo auxiliar os microempreendedores a gerir melhor o seu negócio, mantendo o controle através da automatização e melhor visualização de itens como atendimentos, estoque e fluxo de caixa.

# 1. Introdução

O mercado Brasileiro não vive um momento muito bom, com isso muitos brasileiros são forçados a investir seu dinheiro cautelosamente. Com essa má fase era esperado que o mercado da Beleza e Estética, que na teoria não seria o mais importante para a população em momentos de recessão, deveria estar fechando estabelecimentos, no entanto segundo o Sebrae (2016) o gráfico está indo pro lado oposto. O número de salões de beleza e estabelecimentos relacionados está aumentando cada vez mais, alcançando, em 2016, a marca de 300 mil negócios formalizados nas categorias Micro Empreendedor Individual (MEI), Micro e Pequenas Empresas, sendo cerca de 7 mil salões de beleza abertos mensalmente no território nacional. Se for considerada a grande quantidade de estabelecimentos informais, a quantidade pode passar facilmente de 1 milhão. Pensando nisso, desde 2006 o Sebrae desenvolve ações e projetos para fomentar o empreendedorismo no setor, com o intuito de abrir portas aos empreendedores para eles se especializarem através do conhecimento, tecnologias, inovações e práticas sustentáveis, tornando assim mais fácil o acesso a este mercado tão concorrido que faz a cabeça das pessoas.

Se o mercado está crescendo com uma curva tão acentuada, os empreendimentos irão precisar de ferramentas para gerenciar e manter o controle sobre os atendimentos, carteira de clientes, estoque de produtos, etc. Nestas quase duas décadas do novo milênio, cada vez mais pessoas têm acesso a computadores portáteis e à nuvem. Com isso em mente, surgiu a ideia de criar um sistema hospedado na nuvem, que pudesse ser acessado tanto por desktops quanto por smartphones através de um site responsivo e que combinasse funcionalidades em um único software.

# 2. Objetivos

A mãe do autor possui um salão de beleza, e este artigo tem como objetivo apresentar o desenvolvimento de uma ferramenta gerencial para o empreendimento, que o torne competitivo no mercado através da economia de tempo em várias atividades e da não necessidade de correção de erros humanos, fazendo com que aumente o desempenho e consequentemente a produtividade e lucro. E também por proporcionar uma boa interação com o cliente/usuário através das funcionalidades disponíveis e design atrativo.

# 3. Justificativas

A principal motivação para a escolha do tema deste Trabalho de Conclusão é o fato de os controles do salão serem realizados de forma manual, quando feitos, e foi vista uma possibilidade de construção de um software para controle de atividades, recursos e atendimentos e sobretudo poderia ser personalizado com *features* exclusivas para o negócio dela, e também com a possibilidade de evoluções futuras no sistema.

Como motivo secundário, tem o fato de o autor ter interesse em aperfeiçoar a criação de sites web que proporcionem um uso fácil e intuitivo, além de terem um design que cause uma boa experiência ao usuário.

# 4. Softwares relacionados

Os softwares aqui listados estão mais com o intuito de mostrar os possíveis concorrentes do *7 Beauty* caso ele se torne comercial, por terem funcionalidades bem aproximadas, porém serem pagos, o que não é o objetivo da cliente principal deste software. Dentre eles podemos listar:

* Salão Vip – Sistema bem completo, gratuito apenas para experimentar.
* Trinks – Sistema também bem completo, com várias funcionalidades como marketing do salão. Gratuito para experimentar por 10 dias.

O software usado atualmente no salão se chama “Beleza Soft”, tem várias funcionalidades de gerenciamento como Financeiro, agendamento, no entanto isto apenas no computador da cliente, o que faz com que não seja possível o acesso em outro local ou via smartphone.

Os diferenciais do 7 Beauty estão no fato de ele ser customizado de acordo com as necessidades da cliente e que devido ao seu modo de construção, pode ser administrado pelo cliente no que diz respeito à hospedagem e questões de privacidade. O que traz mais segurança ao consumidor do software pois os dados estão sob seu domínio.

# 5. Metodologia

Como este artigo envolveu o desenvolvimento de um software e ocorreu durante dois semestres do Curso de Análise e Desenvolvimento de Sistemas, o modelo de desenvolvimento escolhido foi o Modelo Cascata, onde são realizadas as tarefas de análise e implementação em tempos diferentes.

Como primeiro passo é realizado o levantamento de requisitos, que é necessário ser feito juntamente com o cliente do software, neste caso a dona do salão de beleza em que será implantado o software. Foram realizadas reuniões com a cliente para alinhar quais seriam as necessidades do software e funcionalidades consideradas imprescindíveis para a mesma exercer sua função de modo que o software consiga auxilia-la com alguns controles que eram de responsabilidade inteiramente dela.

Após estes encontros foram definidos e classificados os requisitos entre funcionais e não funcionais.

## 5.1. Requisitos Funcionais (RF)

Tabela 1. Requisitos Funcionais

|  |  |  |
| --- | --- | --- |
| **Cód.** | **Nome** | **Descrição** |
| **F001** | Cadastro de clientes | Permitir a manutenção de clientes com suas informações como Nome, telefones para contato, endereço, local de trabalho, Nascimento. |
| **F002** | Cadastro de Profissionais | Permitir a manutenção de profissionais do salão, com suas informações como nome, telefones de contato, endereço, função, nascimento, RG e CPF. |
| **F003** | Alertas de Aniversário | Gerar um alerta no sistema, na área de notificações quando um cliente estiver de aniversário. |
| **F004** | Serviços Prestados | Permitir a manutenção de todos os serviços prestados, com seus respectivos preços, nome, tipo (cabeleireiro, manicure e depilação), subtipos (corte, química, tintura, mechas, Hidratação, Cauterização, Progressiva, Selagem, Reconstrução, etc...). |
| **F005** | Valores Serviços | Disponibilizar para os clientes uma tabela com os valores dos serviços. |
| **F006** | Total de Valores dos Serviços Prestados | No ato do agendamento online, o sistema calcula o valor de cada serviço e mostra ao cliente junto ao valor total. |
| **F007** | Cadastros de Fornecedores | Permitir a manutenção dos fornecedores do estabelecimento (Nome Empresa, Nome Consultor, Telefone, Tipos de produtos). |
| **F008** | Pagamento diferente do cobrado | O sistema deve permitir preencher o valor real pago pelo cliente, e assim gerar um registro de débitos/créditos do cliente com o empreendimento. |
| **F009** | Descontos | O Sistema deve permitir dar desconto/definir valor do serviço para o cliente na hora do pagamento. |
| **F010** | Agendamento | O sistema deverá permitir que sejam realizados agendamentos, edições e cancelamentos de atendimentos online. Para realizar marcações de atendimentos, é necessário estar autenticado no sistema. |
| **F011** | Agenda | A agenda deve ser um sistema intuitivo e fácil de usar.  Para visualizar os horários disponíveis na agenda, é necessário estar autenticado no sistema. |
| **F012** | Relatórios | O sistema deve ter uma área para extrair relatórios, desde informações básicas, como números de clientes, atendimentos, profissionais, receitas, despesas e lucro, podendo serem filtrados através de datas, nome do cliente, profissionais, serviços, produtos, horários de atendimento. |
| **F013** | Pagamento | Permitir o pagamento em mais de um método no mesmo atendimento. |
| **F014** | Login | O cliente deve estar autenticado para acessar o sistema. |
| **F015** | Repetição de cliente | O sistema valida se o cliente existe através do cruzamento de informações como nome, sobrenome e telefone (campos obrigatórios). |
| **F016** | Erros Formulários | Quando houver algum erro e retornar ao formulário, auto preencher os campos que o usuário já havia preenchido. |
| **F017** | Histórico de Preços | Manter histórico de valores dos serviços para não afetar o histórico quando houver atualização nos preços. |
| **F018** | Controle de Atendimentos | Permitir o controle dos atendimentos, com informações sobre quais foram os serviços prestados, o preço de cada um, hora do atendimento, desconto, valor pago, custo para o salão, etc... |
| **F019** | Controle de Produtos | Permitir a manutenção dos produtos cadastrados e estoque.  Informações do produto: Separados por tipos (tintas com detalhe de numeração e nome da cor), nome, quantidade (Litros, mililitros, gramas), código de barras (obrigatório, para inserção de produto), estoque mínimo, etc...  O produto deve ter um histórico de compras, com destaque para o último comprado e com qual fornecedor.  O estoque deve gerar um alerta quando estiver menos ou igual o valor do estoque mínimo. |
| **F020** | Controle de Caixa | Controlar receitas e custos, usando uma base de dados alinhada com o RF018. |
| **F021** | Atendimentos | Apenas usuário cliente podem ser atendidos. |
| **F022** | Cadastro de Gerente | Permitir a manutenção de gerentes, Com informações como nome, telefone, CPF, RG, Endereço, Nascimento. |
| **F023** | Motivo de cancelamento | O sistema deverá solicitar ao cliente, o motivo para o cancelamento. Opções apresentadas ao cliente: “Financeiro; Achou preço mais acessível; Tem outro compromisso; Outro (pedir descrição)”.  Opções Gerente e Profissional: mesmas do cliente, mais “Não compareceu; Ligou solicitando o cancelamento”. |

## 5.1. Requisitos Não Funcionais (RNF)

Tabela 2. Requisitos Não Funcionais

|  |  |  |
| --- | --- | --- |
| **Cód.** | **Nome** | **Descrição** |
| **NF001** | Persistência de Dados | Os dados deverão ser armazenados em Banco de Dados SQL. |
| **NF002** | Disponibilidade | O sistema deverá ser hospedado em um local que fique 24h ou aproximadamente este tempo online. |
| **NF003** | Paradigma de Programação | Deverá ser usada uma linguagem Orientada a Objetos. |
| **NF004** | Autenticação | Usuário devem se autenticar com usuário e senha, a qual deve ser criptografada com MD5 antes de sair da máquina do cliente. |
| **NF005** | Responsividade | O sistema deverá ser responsivo, para ser acesso tanto por Desktop quanto aparelhos mobile. |
| **NF006** | Logs | O sistema deve guardar o histórico de movimentações, através de logs. |
| **NF007** | Suporte Web | Permitir o acesso do sistema via internet, e seus dados ficarem armazenados na nuvem. |
| **NF008** | Multi-idioma | O sistema deve ser desenvolvido de modo a estar pronto para receber suporte multi-idioma. O idioma inicial será português, Brasil. |

Durante o desenvolvimento, o cliente acabou por solicitar algumas alterações após experimentar algumas telas e, por exemplo, achou interessante o cliente poder acessar o site sem autenticação, para o mesmo poder ser usado como método de divulgação e se quisesse marcar algo, ele deveria então realizar o login ou registro, caso ainda não o tivesse.

Alterações após ou durante o desenvolvimento de software são corriqueiras e devem ser esperadas pelos gerentes de projeto. Os custos das mudanças costumam ter um alto custo, pois tudo está interligado, e quando se altera alguma parte do software, costumam ter várias funcionalidades afetadas e que voltam a necessitar de atenção por parte dos desenvolvedores, seja para adequar à nova mudança ou corrigir algo que veio a "quebrar" após a execução do sistema. Tendo isso em vista, existem outros modelos de desenvolvimento que são menos custosos com estas mudanças, chamados de métodos ágeis. Com a metodologia ágil as etapas de desenvolvimento são menores, pois o *backlog* é dividido em partes menores, onde são realizados sprints que vão atendendo alguns dos requisitos e fazendo entregáveis com partes do software, para o cliente ir se ambientando e já verificar onde são necessárias mudanças para melhor atendê-lo.

Como já mencionado, não foi escolhido este modo pelo fato de as fases serem bem distintas e com algum período entre elas, sendo considerado o modelo cascata como o que melhor atenderia este *case*.

# 6. Desenvolvimento

## 6.1. Arquitetura

Para arquitetura do software, foram escolhidos componentes que se adequam mais a construção do software e também à familiaridade do autor com tal.  
Para o back-end foram escolhidos a linguagem PHP e o Banco de Dados MySQL. Já par ao front-end foram escolhidos, HTML, CSS, Javascript, e Bootstrap uma tecnologia interessante e que auxilia muito os desenvolvedores a desenvolver layouts responsivos.

## 6.1.1. Hospedagem

Como o sistema é online, como um site, para ele ter disponibilidade *full-time* é aconselhado que seja hospedado na nuvem, através de alguma empresa especializada nisso.

## 6.1.1. Cliente-Servidor

Será usado o modelo cliente servidor na comunicação, onde os clientes que quiserem acessar o sistema, devem acessar o endereço do site (*Uniform Resource Locator, URL*) para se comunicar com o servidor. Quando isso ocorrer, o navegador local envia uma requisição ao servidor remoto, que responde com a página inicial do site, ou no caminho que o usuário solicitou.

## 6.2. Casos de Uso

Foi construída com a ferramenta *Draw.io* os casos de uso para o sistema, levando em consideração os requisitos levantados com o cliente, vistos anteriormente.  
Foi atingido o número de 10 casos de uso principais, sendo: Login, Agendamento, Manutenção de Clientes, Cadastro de Produtos, Cadastro de Serviços, Manutenção de Fornecedores, Extração de Relatórios, Cadastro de Profissionais, Controle de Caixa e Manutenção de Gerente. Isso para um total de 4 atores, sendo eles: Administrador, Gerente, Profissional e Cliente, indo do acesso mais permissivo ao mais restritivo.

O digrama ficou da seguinte forma:

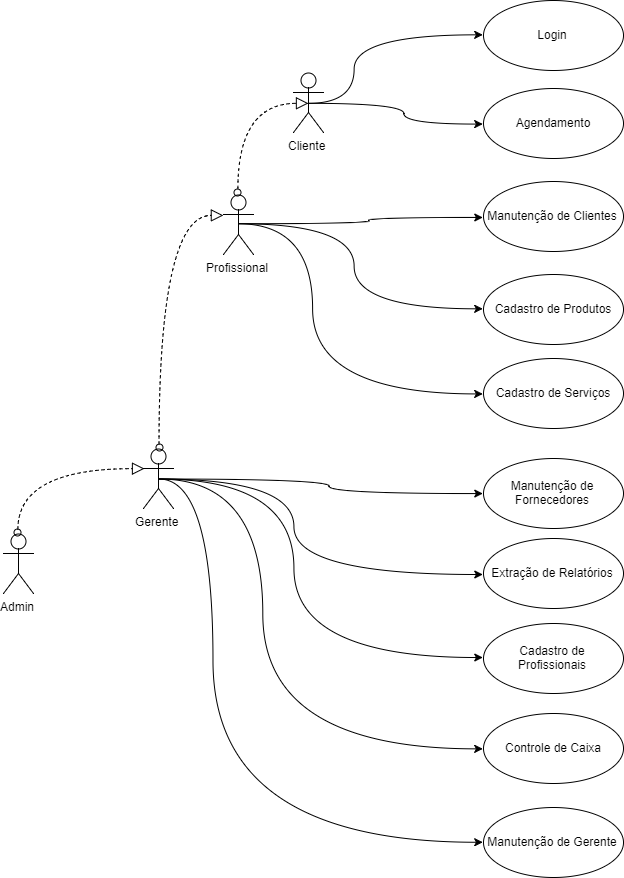


Figura 1. Diagrama de casos de uso

## 6.3. Atores

Abaixo está a tabela com a descrição de cada ator do sistema.

|  |  |
| --- | --- |
| Nome | Descrição |
| Administrador | Terá permissão total dentro do Sistema, único ator que poderá de fato excluir um item (cliente, produto, serviço, fornecedor e profissional), após validar que ele não está sendo usado em lugar nenhum, outros apenas poderão desativar, para não corromper a consistência do banco e outros dados. |
| Gerente | Este usuário irá gerenciar a os profissionais, modificar quaisquer agendas, este usuário também será considerado um profissional, criar/desativar novas contas de profissionais, para criar um gerente é necessário ser um gerente ou usuário com maiores permissões. |
| Profissional | A conta dos profissionais terá acesso a todas as agendas, porém vai poder editar apenas as suas marcações. |
| Cliente | A conta cliente poderá ver todas as agendas, porém não poderá editar nenhuma e verá apenas os horários ocupados, não tendo acesso a informações como cliente que será atendido, serviço, etc. Nas marcações criadas pelo cliente, o mesmo poderá visualizar o serviço e realizar modificações alterar o serviço ou cancelar, mas apenas de datas futuras. |

## 6.4. Banco de Dados (Modelo E-R)

O banco de dados MySQL foi implementado com a entidade relacional a seguir:

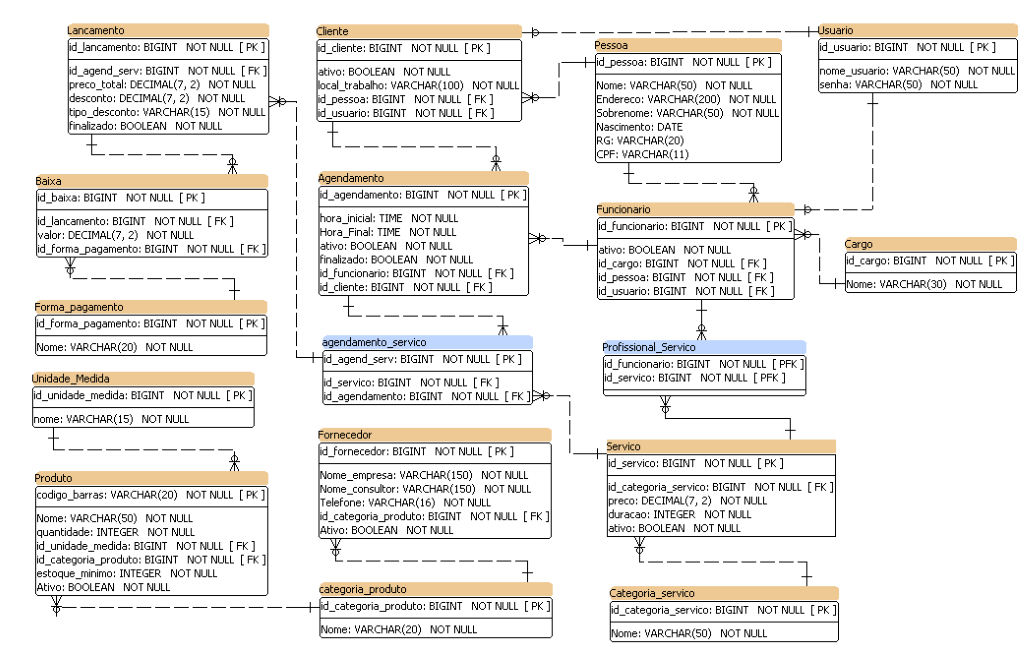


Figura 2. Modelo E-R

O banco de dados é uma estrutura importante e delicada em qualquer sistema, deve ser desenvolvido de uma forma em que consiga armazenar os dados de uma forma organizada e separado em tabelas para cada função e sem perder a performance quando houver um número elevado de informações salvas.

## 6.5. Back-End

Como já dito, o Back-End foi desenvolvido em linguagem PHP, bem conhecida e usada para construir sites no mundo todo. Segundo Wilker Costa, da página Cursos WL Treinamentos Profissionais:

“[..] PHP é uma linguagem de criação de scripts do lado servidor, que foi projetada especificamente para a web. Dentro de uma página HTML, você pode embutir código PHP que será executado toda vez que a página for visitada. O código PHP é interpretado no servidor WEB e gera o HTML necessário para a aplicação rodar.”

Através do uso desta linguagem foram desenvolvidas as seguintes classes para o site:

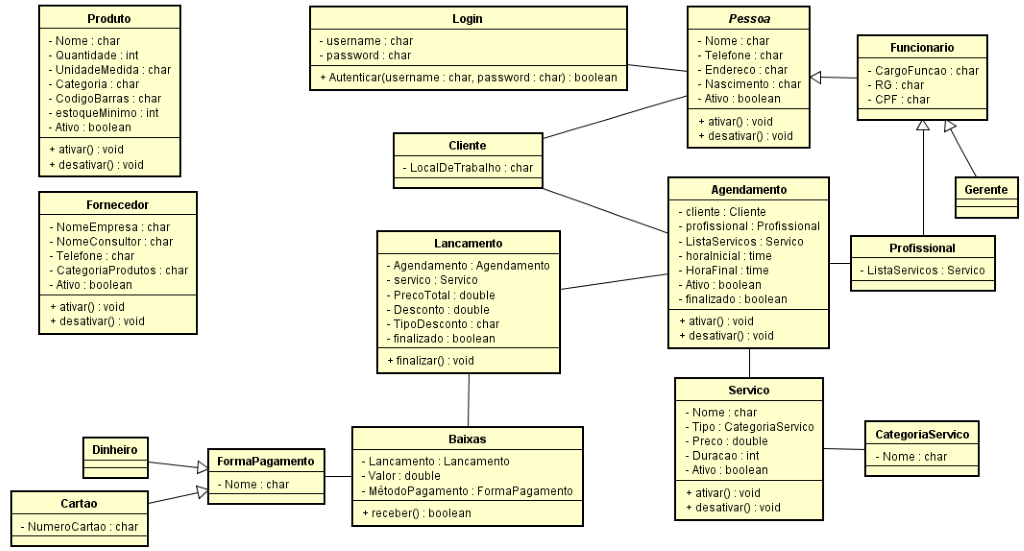


Figura 3. Diagrama de Classes

## 6.6. Front-End

O front-end, por conta de o autor não ter muito domínio de design e layouts, foi a parte mais desafiadora.

Apenas das dificuldades, através do uso do já citado Bootstrap, foi possível chegar em um resultado acima do esperado. Com telas atrativas e responsivas, e com as funcionalidades esperadas pela cliente. Uma das maiores dificuldades neste passo do desenvolvimento foi chegar a um layout que combinasse com o nicho do negócio, pois foram encontradas várias referências de layouts, porém alguns tinham um design antigo ou que não atenderiam às funcionalidades de um modo agradável.

Porém após uma pesquisa mais aprofundada foi encontrado um layout moderno e bonito, que chamaria a atenção dos clientes e deixaria o software mais atrativo aos clientes do estabelecimento. Abaixo seguem imagens de como ficou o modelo de tela de login e página inicial.

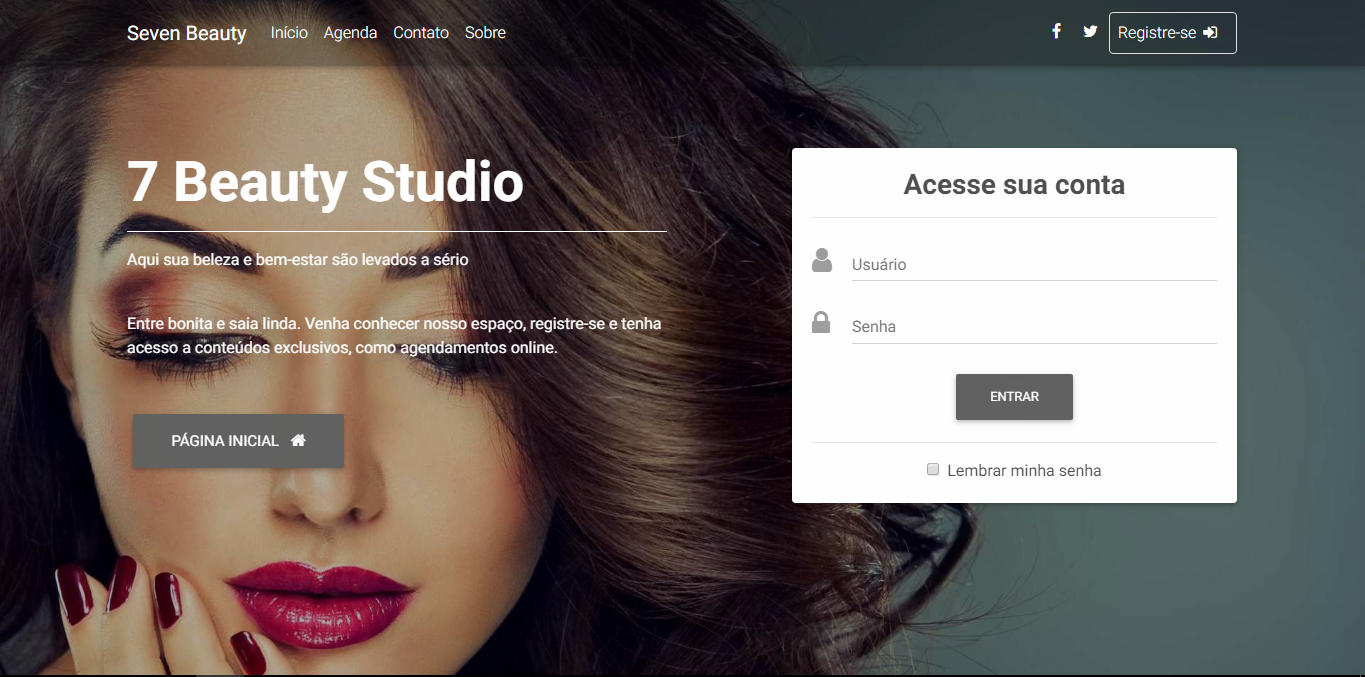


Figura 4. Captura da tela de login

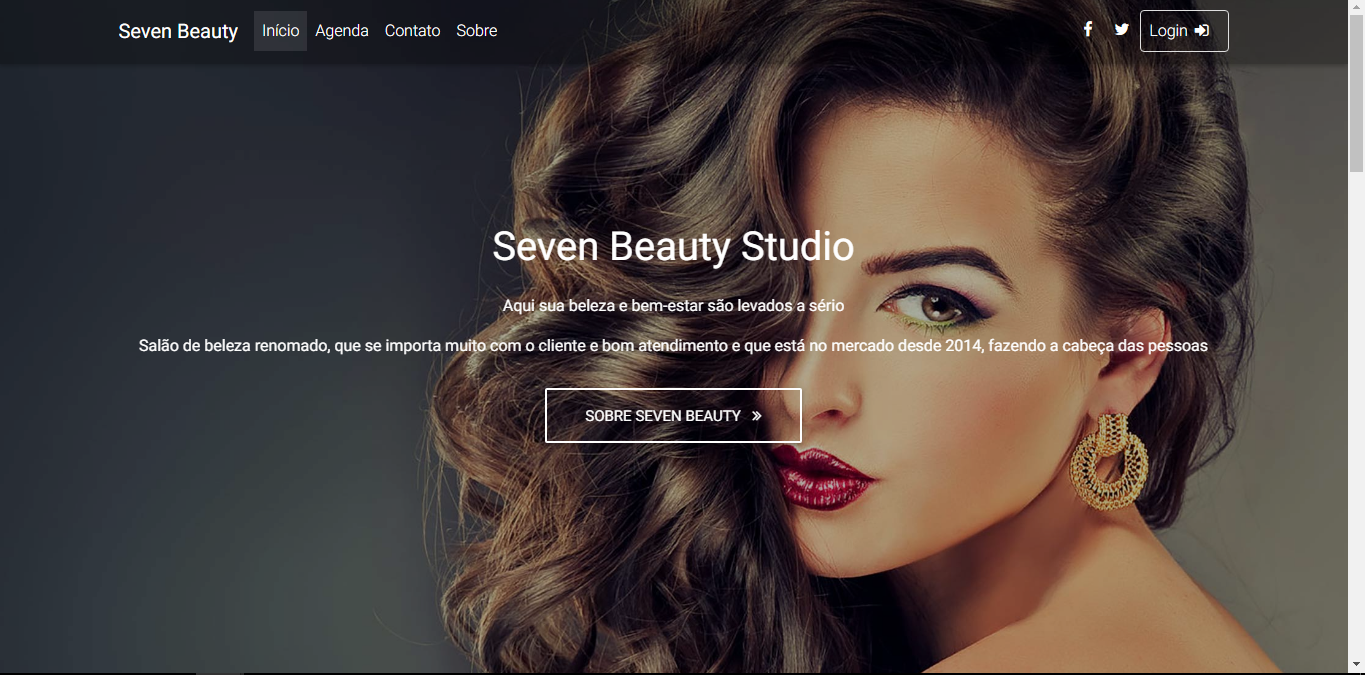


Figura 5. Captura da tela inicial

# Referências

VALE A PENA MONTAR UM SALÃO DE BELEZA? SEBRAE. Disponível em: <<http://www.sebrae.com.br/sites/PortalSebrae/artigos/vale-a-pena-montar-um-salao-de-beleza,efb8d62b2b886410VgnVCM1000003b74010aRCRD>>. Acesso em: 14 maio 2018.

Talma, Thiago Moreira (2006) Em Desenvolvimento de Software de Auxílio ao Fluxo e ao Compartilhamento de Informações Administrativas em Ambientes Empresariais [Juiz de Fora] 2006

Batista, Riann Martinelli (2015) Em Proposta de desenvolvimento de software para gestão de viagens da DEAD/UFVJM – SIGEV

WANT TO BE AGILE? STOP DOING "PROJECTS" AGILE METHODOLOGY. Disponível em: <<http://agilemethodology.org/>>. Acesso em: 14 maio 2018.

PHP – O QUE É E COMO FUNCIONA? CURSOS WL Disponível em: <[https://www.cursos.wlconsultoria.net/blog/php-o-que-e-e-como-funciona](https://www.cursos.wlconsultoria.net/blog/php-o-que-e-e-como-funciona/)>. Acesso em: 14 maio 2018.