

FACULDADE UNIRB FORTALEZA GRADUAÇÃO TECNOLÓGICA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

GABRIEL RODRIGUES LIMA FONSECA

AGENDIT - SISTEMA DE LOCAÇÃO DE HORÁRIOS

GABRIEL RODRIGUES LIMA FONSECA

AGENDIT - SISTEMA DE ALOCAÇÃO DE HORÁRIOS.

Trabalho de conclusão de curso apresentado para a Faculdade Unirb Fortaleza - como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas sob orientação do (a) Prof (a). Esp. Leandro Moreira Lopes

Fortaleza - CE Junho/2021 Fonseca, Gabriel Rodrigues Lima.

AgendIT - Sistema de locação de horários / Gabriel Rodrigues Lima Fonseca.

Fortaleza Çe, 2021.

45 f.

Monografia (graduação) do Curso de Tecnólogo em Análise e Desenvolvimento de Sistemas. – Faculdade <u>Unitb</u> Fortaleza.

Orientação: Prof. Leandro Moreira Lopes.

Uml. 2. Tecnologia. 3. Automação. I. Título.

CDD: 005

Trabalho apresentado como requisito necessário para a obtenção do título de
Tecnólogo em Análise e Desenvolvimento de Sistemas pela Faculdade Unirb. Qualquer
citação atenderá às normas da ética científica.

Gabriel Rodrigues Lima Fonseca

Trabalho apresentado em 30/06/2021

Orientador (a) Prof (a). Esp Leandro Moreira Lopes

1o. Examinador (a) Prof (a). Fábio

2o. Examinador (a) Prof (a). Igor

Coordenador (a) Prof (a). Dra. Diana Cajado

Dedico tal trabalho para meus pais que apoiaram o ingresso na faculdade, aos familiares e também a Deus! Agradeço ao professor e coordenador do curso de Análise e Desenvolvimento de Sistemas, Prof. Leandro.

RESUMO

Este documento tem como objetivo descrever problemas e soluções em uma rotina bem cotidiana de um adulto: agendar um horário. Utilizando uma empresa do ramo de estética, relatando problemas de ambos os lados, cliente e empresa, propondo soluções e explicando-o caminhos para atingir essa solução. Através dessas possíveis soluções foi-se desenvolvido um projeto com todas as etapas necessárias para seu desenvolvimento e implantação: banco de dados, tecnologias adotadas no desenvolvimento, diagramações UML, etc. Apresentando-se como artigo, através de 43 páginas descritas conforme as normas exigidas pela ABNT (Associação Brasileira de Normas Técnicas), buscou-se no artigo representar também uma documentação exigida para um software bem desenvolvido, referindo-se em tópicos técnicos e diagramados.

ABSTRACT

This document aims to describe problems and solutions in an adult's everyday routine: schedule a time. Using a company in the field of aesthetics, reporting problems from both sides, client and company, proposing solutions and explaining ways to achieve this solution. Through these possible solutions, a project was developed with all the necessary steps for its development and implementation: database, technologies adopted in the development, UML diagrams, among others. Presenting itself as an article, through 43 pages described according to the standards required by the ABNT (Brazilian Association of Technical Standards), the article also sought to represent a documentation required for a well-developed software, referring to technical topics and diagrams.

SU			•
€ 111	N/I /N		1
JUI		\mathbf{r}	,

	1. INTRODUÇÃO)
--	---------------	---

1.1. DESCRIÇÃO DO TEMA
1.2. JUSTIFICATIVA
1.3. OBJETIVOS
1.4. ANÁLISE DO AMBIENTE ORGANIZACIONAL
1.4.1. Identificação da empresa
1.4.2. Ramo de atividade
1.4.3. <u>Organograma</u>
1.4.4. Plataforma tecnológica necessária
1.5. MODELAGEM DE PROCESSOS DE NEGÓCIO DA ORGANIZAÇÃO
1.5.1. Cadeia de valor e processo crítico
1.5.2. Processo AS-IS
1.5.3. Cenário atual
1.5.4. Proposta de melhoria
1.5.5. <u>Processo TO-BE</u>
1.6. LEVANTAMENTO DO SISTEMA
1.6.1. Descrição do minimundo do sistema
1.6.2. Detalhamento dos processos de negócio (diagrama de atividades)
1.6.3. Requisitos do sistema
1.6.3.1. Requisitos funcionais
1.6.3.2. Requisitos não funcionais
1.6.4. <u>Descrições das regras de negócio</u>
1.7. METODOLOGIA DE DESENVOLVIMENTO
1.8. CRONOGRAMA DE CONFECÇÃO
1.9. ORÇAMENTO DO TRABALHO
1.9. ORÇAMENTO DO TRABALHO
2. MODELO DE CASOS DE USO
2.1. RELAÇÃO DOS CASOS DE USOS
2.1. RELAÇÃO DOS CASOS DE USOS
2.2. DESCRIÇÃO DOS ATORES
2.2. DESCRIÇÃO DOS ATORES 2.3. DIAGRAMA DE CASO DE USO 2.4. DESCRIÇÕES DOS CASOS DE USO 3. MODELO DE CLASSES 3.1. DIAGRAMA DE CLASSES 3.2. DICIONÁRIO DE CLASSES 4. MODELO DE INTERAÇÕES 4.1. DIAGRAMA DE SEQUÊNCIAS 5. MODELO DE TRANSIÇÃO DE ESTADOS 5.1. DIAGRAMA DE TRANSIÇÃO DE ESTADOS
2.2. DESCRIÇÃO DOS ATORES 2.3. DIAGRAMA DE CASO DE USO 2.4. DESCRIÇÕES DOS CASOS DE USO 3. MODELO DE CLASSES 3.1. DIAGRAMA DE CLASSES 3.2. DICIONÁRIO DE CLASSES 4. MODELO DE INTERAÇÕES 4.1. DIAGRAMA DE SEQUÊNCIAS 5. MODELO DE TRANSIÇÃO DE ESTADOS 5.1. DIAGRAMA DE TRANSIÇÃO DE ESTADOS
2.2. DESCRIÇÃO DOS ATORES 2.3. DIAGRAMA DE CASO DE USO 2.4. DESCRIÇÕES DOS CASOS DE USO 3. MODELO DE CLASSES 3.1. DIAGRAMA DE CLASSES 3.2. DICIONÁRIO DE CLASSES 4. MODELO DE INTERAÇÕES 4.1. DIAGRAMA DE SEQUÊNCIAS 5. MODELO DE TRANSIÇÃO DE ESTADOS 5.1. DIAGRAMA DE TRANSIÇÃO DE ESTADOS 6. PROJETO DE BANCO DE DADOS 6.1. PROJETO LÓGICO DE BANCO DE DADOS
2.2. DESCRIÇÃO DOS ATORES 2.3. DIAGRAMA DE CASO DE USO 2.4. DESCRIÇÕES DOS CASOS DE USO 3. MODELO DE CLASSES 3.1. DIAGRAMA DE CLASSES 3.2. DICIONÁRIO DE CLASSES 4. MODELO DE INTERAÇÕES 4.1. DIAGRAMA DE SEQUÊNCIAS 5. MODELO DE TRANSIÇÃO DE ESTADOS 5.1. DIAGRAMA DE TRANSIÇÃO DE ESTADOS 6. PROJETO DE BANCO DE DADOS 6.1. PROJETO LÓGICO DE BANCO DE DADOS 6.2. PROJETO FÍSICO DE BANCO DE DADOS
2.2. DESCRIÇÃO DOS ATORES 2.3. DIAGRAMA DE CASO DE USO 2.4. DESCRIÇÕES DOS CASOS DE USO 3. MODELO DE CLASSES 3.1. DIAGRAMA DE CLASSES 3.2. DICIONÁRIO DE CLASSES 4. MODELO DE INTERAÇÕES 4.1. DIAGRAMA DE SEQUÊNCIAS 5. MODELO DE TRANSIÇÃO DE ESTADOS 5.1. DIAGRAMA DE TRANSIÇÃO DE ESTADOS 6. PROJETO DE BANCO DE DADOS 6.1. PROJETO LÓGICO DE BANCO DE DADOS

7. PROJETO DE INTERFACE GRÁFICA
7.1. HIERARQUIA DAS TELAS
7.2. MAPA DE NAVEGAÇÃO
7.3. PADRONIZAÇÃO DE BOTÕES, ÍCONES E OUTROS ATALHOS
7.4. LAYOUT DAS TELAS E RELATÓRIOS
8. <u>PROJETO DE IMPLANTAÇÃO</u>
8.1. PROCEDIMENTOS DE IMPLANTAÇÃO
8.2. AMBIENTE OPERACIONAL NECESSÁRIO E A SEQUÊNCIA COM QUE OS
MÓDULOS DEVERÃO SER IMPLANTADOS
8.3. ESTRATÉGIA DE IMPLEMENTAÇÃO
9. AMOSTRA DO SISTEMA
10. CONCLUSÕES E TRABALHOS FUTUROS
11. REFERÊNCIAS BIBLIOGRÁFICAS

1. INTRODUÇÃO

Com a vida moderna em avanço e o tempo sendo mais curto para todos nós, economizar tempo em coisas do cotidiano como por exemplo agendar um horário em um salão de beleza é algo interessante, pois todo tempo hoje em dia é precioso. Com isso em mente, pensei em um sistema que pudesse unir a praticidade de algo online com a rapidez de algo automatizado, por isso foi criado o AgendIT - Sistema de locação de horários; o sistema consta com gerenciamento de agendamentos, relatórios de clientes, cadastros de contas, validações de registro, entre outros.

Apesar de já existir softwares similares, o mercado carece de sistemas baratos que cumpram o básico, logo, o AgendIT vem com essa proposta de baixo custo.

1.1. DESCRIÇÃO DO TEMA

Nos dias de hoje, onde anda tudo tão corrido, ter que fazer uma ligação para marcar um horário, algo tão simples e que na prática não deveria ser demorado, acaba sendo uma dor de cabeça; motivos como: falta de esclarecimento por parte da recepcionista, horários condizentes com a sua disponibilidade etc. Estes fatores podem levar a uma ligação demorada, consumindo um tempo precioso do seu dia, além disso pode acabar ocorrendo um agendamento duplicado, erro humano, entre outros assuntos que causariam aborrecimento de ambas as partes.

1.2. JUSTIFICATIVA

Nos dias atuais, a correria da cidade grande exige tempo. Automatizar uma marcação de horário em um salão de beleza, seja através de um aplicativo mobile ou até mesmo via web (browser - Desktop/Notebook), é necessário. Transformar esse processo que por diversos motivos possa ser demorado, agrega valor ao estabelecimento que irá fornecer este tipo de serviço, dando uma satisfação maior ao cliente e digitalizando seus processos, além da redução de custos.

1.3. OBJETIVOS

Portanto, para solucionar tais problemas, necessitamos levantar maneiras de corrigir estes processos e para isso pensamos e desenvolvemos o AgendIT, o AgendIT tem como objetivo melhorar e automatizar esse relacionamento (cliente-estabelecimento), deixando de forma clara todas as informações necessárias (datas disponíveis, como principal exemplo) e promovendo um acerto entre o cliente e o estabelecimento de modo rápido e prático.

1.4. ANÁLISE DO AMBIENTE ORGANIZACIONAL

A análise de ambiente influencia diretamente na evolução da empresa. Com isso em mente, podemos dizer que um ambiente bem-organizado só gera frutos positivos para a empresa, portanto, fazer uma análise do ambiente organizacional é de suma importância para uma empresa que vise crescimento contínuo e um ambiente harmônico.

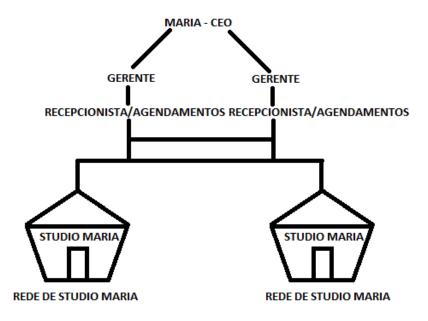
1.4.1. Identificação da empresa

O sistema abrange diversos segmentos que utilizam da marcação de horários para gerir seus atendimentos, sendo assim, o ramo de uma possível empresa que venha a adquirir os direitos do sistema seria uma empresa de salão/studio de beleza; um exemplo seria a rede de beleza Studio Maria, uma rede do ramo de beleza que se localiza no estado do Ceará, tendo por sua vez a proprietária Maria, que dá nome à rede.

1.4.2. Ramo de atividade

O Studio Maria trabalha diretamente com cortes, limpeza e outros procedimentos capilar e estéticos, sendo assim, de forma generalizada ele tem um ramo de negócio que denominamos de clínica de estética.

1.4.3. Organograma da empresa



1.4.4. Plataforma tecnológica necessária

O sistema foi desenvolvido para ser utilizado via browser, ou seja, é uma aplicação web; portanto, para a utilização tanto cliente quanto administrador, será necessário acesso através do navegador seja desktop ou mobile, além de um acesso à Internet.

Será utilizado um banco de dados em nuvem, pois as informações ficarão armazenadas em uma hospedagem na internet, tornando o sistema disponível para usuários. O sistema foi escrito com a linguagem C#, utilizando de ORM Dapper e para a sua interface gráfica, foi utilizado a biblioteca do JavaScript chamada React JS.

1.5. MODELAGEM DE PROCESSOS DE NEGÓCIO DA ORGANIZAÇÃO

A modelagem de processos de negócio é uma ferramenta utilizada para entender os fluxos de trabalho da empresa. (**Sebrae Alagoas** em abril 8, 2020).

Um bom sistema começa pela sua modelagem, através da modelagem é por onde ele vai tomando forma e se criando, portanto, discorrer sobre é uma etapa importante para um projeto bem desenvolvido, entender os fluxos da empresa, como se trabalha cada processo etc. É de suma importância entender a modelagem para ter um progresso dentro do projeto.

1.5.1. Cadeia de Valor e Processo Crítico

O atual processo é tem um enorme valor na empresa, pois através dele que é adquirido uma das satisfações do cliente: o atendimento. Ai está também o nosso maior processo crítico, pois o atendimento hoje em dia, em praticamente todo canto, é levado em consideração para um possível retorno do cliente, afinal, não adianta ter um "excelente corte de cabelo" se você nem sequer consegue agendar um horário.

Como este processo é divido em dois módulos atualmente: agendamento por ligação e também por WhatsApp, temos dois tipos de pontos críticos, sendo eles:

No primeiro ponto: uma ligação duradoura, com diversas etapas e perguntas apenas para marcar um agendamento é meio chato para o cliente, pois está tomando-o seu tempo que por muitas vezes é precioso, fora erros que um atendimento humano estar suscetível a acontecer.

No segundo ponto: um atendimento pelo WhatsApp é um pouco menos custoso, porém, caso o cliente não seja atendido rapidamente poderá causar algum desgosto referente ao atendimento, além de ser necessário uma boa escrita, pois não é todo mundo que gosta de ter um atendimento com diversos erros de português.

Esses dois pontos do atual processo de alguma maneira, supre, porém, há percas que poderiam agregar bem mais na receita, a satisfação do cliente, portanto, automatizar este processo excluindo o contato humano diretamente pode ser uma possível opção para aumentar o sucesso do atendimento.

1.5.2. Processo AS-IS

Atualmente, o processo de agendamento funciona da seguinte maneira: o possível cliente entra em contato com alguma loja credenciada da rede de 2 maneiras possíveis: contato via ligação telefônica ou via Whatsapp.

Um contato via ligação telefônica é um processo mais lento e leva a um custo alto sobre o telefone, pois precisa de alguns minutos para o agendamento ser efetivado. Além disso, ainda é necessária uma pessoa responsável, que fica trabalhando quase integralmente para realizar esses atendimentos.

Já via Whatsapp, não há necessidade de um atendente em tempo integral para respondê-lo, por não é tão síncrono. Outra vantagem é que as mensagens são enviadas pela internet, utilizando-se o wifi da unidade, diminuindo ainda mais o custo. Mas mesmo com essas vantagens, ainda é um trabalho de certa forma manual, pois precisa haver essa resposta em um espaço de tempo viável para não perder a marcação, configurando um contato rápido entre cliente/atendente e precisando que alguém fique disponível quase todo o período.

1.5.3. Cenário atual

Atualmente esse processo de agendamento de horário é feito de forma menos automatizada e uma interação humana maior; há primeiro a ação do cliente em ter que pegar seu celular e ligar para alguma loja da rede, isso já é um gasto do cliente, pois a ligação requer créditos sejam pré ou pós-pago, caso nenhuma linha esteja ocupada a atendente irá atender a chamada e organiza através de planilha no Excel.

O método via Whatsapp é similar, porém, um pouco mais demorado, caso não tenha ninguém para monitorá-lo no momento: há o contato para algum dos estabelecimentos da rede, ao receber o contato através do Whatsapp a atendente irá trocar informações até agendar o melhor horário para o cliente e nisso há um gasto de tempo, onde a funcionária poderia tá alocada em alguma outra atividade que gerasse um retorno maior para a instituição.

Recentemente, foi-se percebido por nós, diretores da empresa, que esses cenários acabam desgastando os nossos clientes por diversos motivos: erro humano, duplicidade de agendamento, erro na hora de agendar, entre outros... esses erros acabam acarretando em perca de receita para a empresa.

1.5.4. Proposta de melhoria

Temos como proposta de melhoria, através do sistema, sanar lacunas deixadas pela falta de organização do processo atual da empresa, empregando na cultura de agendamentos uma maior automatização e garantindo uma experiência satisfatória. Há alguns questionamentos sobre abrangência de relatórios, visto que através do Excel temos como gerir essas informações; o software contará com alguns relatórios iniciais, podendo ser incrementado novos tipos de relatórios ao decorrer do tempo, essas informações sendo extraídas da base.

1.5.5. Processo TO-BE

O processo será de forma simples e não tão extensa. Será disponibilizado um acesso - admin e cliente - onde através desse portal será possível o cliente fazer o cadastro, login e escolher um agendamento para ele, sendo esse último o diferencial do antigo processo, em que no caso o cliente teria que ligar, falar com atendente, marcar manualmente em uma planilha; do lado do admin, será disponibilizado relatórios e aceitar agendamentos, entre outras funções do sistema. Esses relatórios teriam que ser criados via Excel no antigo processo, mas agora, com a aplicação proposta, basta apenas um clique e você tem esse relatório disponibilizado.

1.6. LEVANTAMENTO DO SISTEMA

Para fazer um bolo você necessita de ingredientes e etapas, quando esse bolo é para um cliente você também necessita dos ingredientes que o cliente pediu; com a criação de um sistema não é diferente, portanto, uma das etapas mais importantes é o levantamento do sistema, onde você irá ver através dos requisitos o necessário para conseguir desenvolver um sistema que supra as necessidades seguindo suas etapas.

Entender aquilo que o cliente deseja ou o que o cliente acredita que precisa e as regras do negócio ou processos do negócio. (Ricardo Veríssimo, novembro de 2007).

1.6.1. <u>Descrição do minimundo do sistema</u>

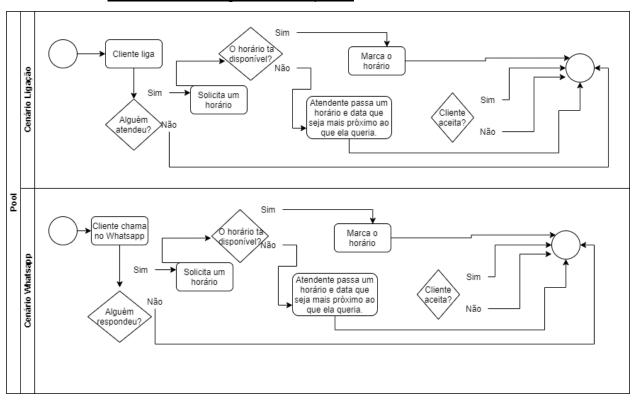
Este sistema foi desenvolvido com o intuito de agilizar os agendamentos de qualquer estabelecimento que se enquadre nos requisitos dele, um sistema web, podendo ser acessado de qualquer aparelho que tenha conexão com Internet. Basicamente, os requisitos como: módulos de login, módulo de agendamento, relatórios,

devem estar inseridos no sistema. Com isso em mente, desenvolvemos um sistema contando com um processo de agendamento independente de horário e é aceito através do usuário administrador, ou seja, o recepcionista, agilizando esse processo de agendamento.

O banco, em Nuvem, recebe a informação do cliente (nome, seu id de cliente, e-mail) e o horário que ele deseja, isso através da nossa API, ao fazer essa solicitação, o usuário administrador consegue ver essas informações de solicitação através da guia solicitações de horários - presente no navbar do site - após o aceite, o cliente recebe um e-mail informando desse aceite.

O sistema ainda dispõe de relatórios de clientes, usuários, podendo ser incrementado outros relatórios, como por exemplo um relatório de custos, lucros por mês, entre outros.

1.6.2. Processos de negócio da empresa



1.6.3. Requisitos do sistema

Em outras palavras, quando falamos de requisitos, tange a solicitação do cliente, como: necessidades, exigências e desejos (Danielle Teixeira, 2018) - Basicamente, os requisitos do sistema são exigências que o cliente pede e que se enquadram nas regras de negócios do sistema.

1.6.3.1. Requisitos Funcionais

- Deve efetuar o cadastro do cliente e admin
- Deve mostrar relatório de cliente somente para o usuário admin
- Deve ter verificação de login e senha corretamente
- Deve enviar e-mail enviando dados de login
- Ao administrador, deve-se aceitar ou recusar, caso não, enviar um e-mail ao admin padrão avisando quantos há de solicitações não atendidas.

1.6.3.2. Requisitos não funcionais

- Acesso à Internet
- Desktop/Notebook ou celular para efetuar o acesso.

1.6.4. Descrições das regras de negócio

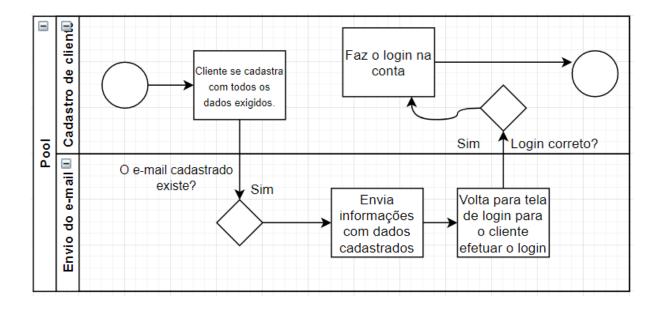
- Tanto cliente quanto administrador devem ter acessos ao sistema.
- Foi desenvolvido uma tela de login e registrar, contendo opção de cliente ou admin.
 - Há uma necessidade de agilidade nos agendamentos.
- Para tal, quando o cliente efetuar o agendamento o admin padrão ou a loja receberá um e-mail com o e-mail de quem solicitou e as informações da solicitação, alertando sobre a solicitação de agendamento pendente.
 - Necessidade de relatórios
- De início, somente um relatório foi desenvolvido, entretanto, outros relatórios serão feitos ao decorrer do tempo.
- Você só pode criar conta de um único tipo, não podendo ser uma mesma conta do tipo ADMIN e do tipo CLIENTE

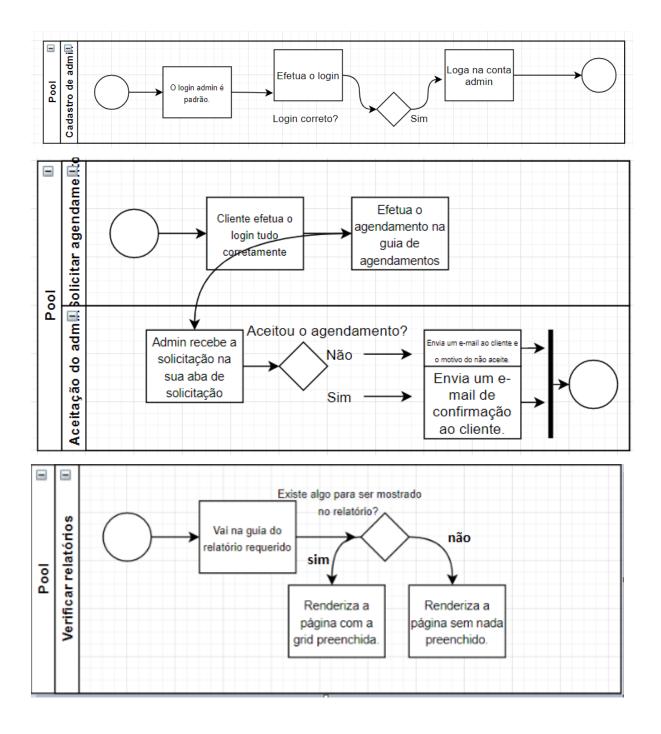
 Ao criar a conta, há dois checkbox que decide se você é administrador ou cliente, quando um é clicado o outro é bloqueado.

1.7. METODOLOGIA DE DESENVOLVIMENTO

Através de uma entrevista com pessoas internalizadas nos processos da empresa, resolvemos levantar os requisitos. Efetuamos cerca de 3 dias de entrevistas, com o CEO, Gerentes e funcionários do cotidiano da empresa, que estão mais próximos ao processo avaliado, no seu dia a dia.

Com essa entrevista, conseguimos levantar os requisitos iniciais do sistema; partimos depois para ver o melhor nível de tecnologia, que se enquadraria de uma maneira geral para a empresa e também outras empresas que pudéssemos vender o sistema. Com isso em mente, levantamos a tecnologia que é fácil desenvolver e dá um suporte legal para trabalharmos: C# .Net, trabalhando com versões mais atuais do .NET, ORM Dapper para fazermos a persistência dos dados e para front, decidimos trabalhar com a biblioteca JavaScript REACT.JS; banco de dados decidimos trabalhar com o SGBD Sql Server e colocá-lo em cloud. Além disso, para ajudar a documentar também utilizamos o Swagger.





1.8. CRONOGRAMA DE CONFECÇÃO

Para iniciar, o designer irá iniciar com o desenho da interface visual, trabalhando com as telas iniciais, logo após o desenvolvedor JR junto comigo iremos desenvolver questões de banco de dados e desenvolvimento da API e o Desenvolvedor React.JS irá iniciar o front. Contando com o início de trabalho do Designer, iniciando com a modelagem do sistema, o cronograma de confecção terá 3 meses, sendo cada mês um ponto de avanço:

	CRONOGRAMA	DE CONFECÇÃ	(O	
1º MÊS	MODELAGEM D	E DADOS E INÍC	CIO PROTÓTIPO	DESIGNER
2º MÊS	DESENVOLVIM	ENTO DO FRON	T, BACK E SCRIF	PTS DE DADOS
3º MÊS	FINALIZAR DES	ENVOLVIMENT	D, TESTES E EN	TREGA.

Cronograma

1.9. ORÇAMENTO DO TRABALHO.

Apesar de ser uma sugestão de aplicação para resolver um problema comum no cotidiano de empresas de saúde, estimamos o gasto em cima dos valores de mercado dos profissionais necessários e o cronograma apresentado na seção anterior.

	ORÇAMENTO D	O TRABALHO	R\$
1 - Desenvolved	or JR C# .NET		R\$ 2,300 MÊS
1 - Designer			R\$ 2,000 MÊS
1 - Desenvolved	or React.JS		R\$ 3,000 MÊS
5 - Computadore	s I5 16GB RAM	SSD 520GB	R\$ 4,000 UN
		TOTAL	R\$ 41900

Orçamento previsto

Orçamento feito com base nos 3 meses do projeto.

2 - MODELO DE CASOS DE USO

Quando falamos de modelo de caso de uso, estamos falando de um **modelo** que descreve como os usuários interagem com o sistema para resolver um problema.

O Diagrama de Caso de Uso serve para representar como os casos de uso interagem entre si no sistema e com os usuários (atores), ou seja, como as funcionalidades se relacionarão umas com as outras e como serão utilizadas pelo usuário, durante o uso do sistema. (Plínio Ventura, Fevereiro de 2016).

2.1. RELAÇÃO DOS CASOS DE USOS

CSU 01 - Registrar: Para fazer o registro da sua conta no sistema, persistir a camada de dados.

CSU 02 - Login: o login será para acessar ao sistema, tendo como princípio o nível de usuário de cada um.

CSU 03 - Ver relatório: relatório de clientes, financeiros, etc, somente o admin terá acesso a essa funcionalidade.

CSU 04 - Agendar horário: para o cliente fazer o agendamento do horário desejado.

CSU 05 - Aceitar/Recusar horário: para o cliente poder ter o horário agendado, será necessário o admin aceitar. Somente o admin poderá ter esse privilégio.

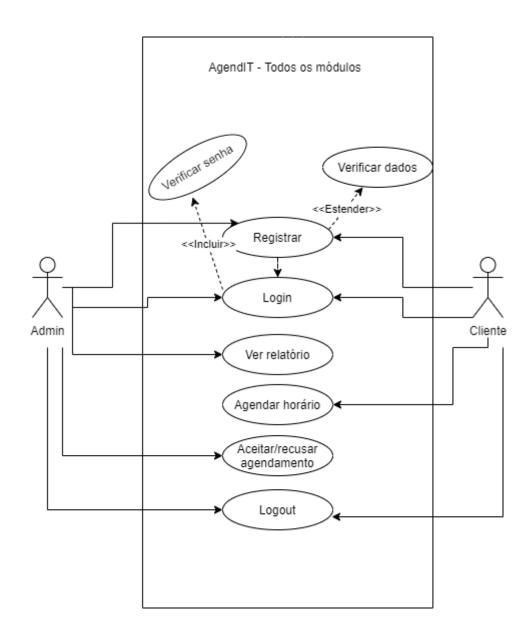
CSU 06 - Logout: para sair do sistema.

2.2. DESCRIÇÃO DOS ATORES

Cliente/Usuário: o Cliente será literalmente, o cliente, terá acesso somente a funcionalidades ao seu nível de cliente.

Admin: é o administrador do sistema, o funcionário da loja, além de algumas funções que o cliente também tem acesso, o admin terá relatórios, permissões a mais que o cliente, etc.

2.3. DIAGRAMA DE CASO DE USO.



2.4. DESCRIÇÕES DOS CASOS DE USO

CSU01 - REGISTRAR

- Sumário: Para fazer o registro no sistema, persistir a camada de dados.
- Ator primário: Cliente/Admin
- Ator secundário: N/A
- Pré-condições: Ter um e-mail não cadastrado
- Fluxo principal: 1. Cliente preenche os dados;
 - 2. Recebe um e-mail confirmando a inscrição;
 - 3. Tela volta para a tela de login;
 - 4. Se loga com os dados passado no registro;
- Fluxo alternativo: N/A
- Garantia de sucesso: E-mail de confirmação de cadastro recebido no e-mail
- Requisitos especiais: N/A

CSU02 - LOGIN

- Sumário: Para logar no sistema.
- Ator primário: Cliente/Admin
- Ator secundário: N/A
- Pré-condições: Ter uma conta no sistema
- Fluxo principal: 1. Põe os dados nos campos de login e clica para registrar
- Fluxo alternativo: N/A
- Garantia de sucesso: Ter conseguido acessar as telas do sistema.
- Requisitos especiais: N/A

CSU03 - RELATÓRIOS

- Sumário: Acessar tela de relatórios dos clientes
- Ator primário: Admin
- Ator secundário: N/A
- Pré-condições: Ter logado com usuário admin.
- Fluxo principal: 1. Logar no sistema como admin; 2. Clicar no botão Clientes e ter a grid preenchida pelos dados.
- Fluxo alternativo: N/A
- Garantia de sucesso: Acessar a tela de relatórios de clientes e receber dados nele.
- Requisitos especiais: N/A

CSU04 – AGENDAR HORÁRIO

- Sumário: Para cliente fazer a solicitação do horário agendado
- Ator primário: Cliente
- Ator secundário: N/A
- Pré-condições: Ter logado com usuário cliente.
- Fluxo principal: 1. Logar no sistema como admin;
 2. Clicar em horário desejado e após no botão solicitar.
- Fluxo alternativo: N/A
- Garantia de sucesso: Quando aparecer confirmação de envio da solicitação
- Requisitos especiais: N/A

CSU05 - RESPONDER AGENDAMENTO (ACEITAR/RECUSAR)

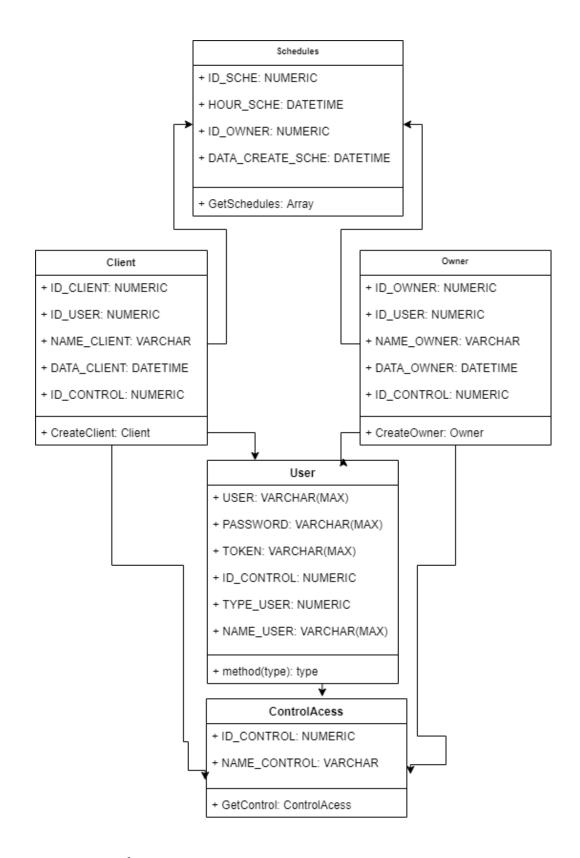
- Sumário: Momento onde o admin aceita ou recusa o horário;
- Ator primário: Admin
- Ator secundário: N/A
- Pré-condições: Está logado no sistema em modo admin e ter solicitações pendentes de aceite.
- Fluxo principal: 1. Logar no sistema como admin;
 - Clicar em schedules e efetuar o aceite ou não no botão verde e vermelho respectivamente
- Fluxo alternativo: N/A
- Garantia de sucesso: Quando aparecer confirmação de envio da solicitação
- Requisitos do sistema: N/A

3 - MODELO DE CLASSES

O modelo de classes serve para demonstrar a nível de relacionamento entre tabelas, fazendo uma analogia: demonstrar como uma pessoa se comunica com a outra para poder chegar na pessoa final. No meu sistema, por ser desenvolvido usando um banco relacional e trabalhando com orientado a objetos, o modelo de classes está totalmente envolvido, pois há vários relacionamentos dentro do meu sistema para que consiga suprir as regras de negócios requisitadas.

Esse tipo de representação é bastante útil, pois serve no desenvolvimento de sistemas, definindo todas as classes que o sistema precisa ter e serve como uma forte base para a construção de outros diagramas que definem o tipo de comunicação, sequência e estado do sistema. (Carlos Barcelos, novembro de 2019)

3.1. DIAGRAMA DE CLASSES



3.2. DICIONÁRIO DAS CLASSES

ControlAcess - Classe de controle de acesso, com duas classes padrões: ADMIN & Cliente, foi criada para caso haja outro tipo de acesso ao sistema.

User - Em uma relação 1 para N, a classe USER foi criada pensando nisso, um usuário pode ter várias contas, mas uma conta só é ligada somente a um usuário.

Client - Para quem criou conta a nível cliente, para marcar horários.

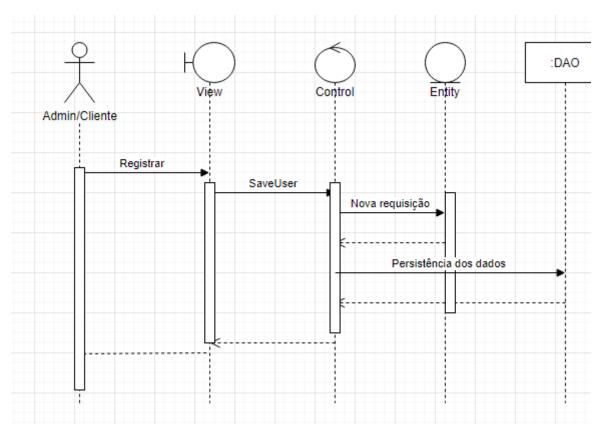
Owner - Para contas do nível admin.

Schedules - Marcações de horários.

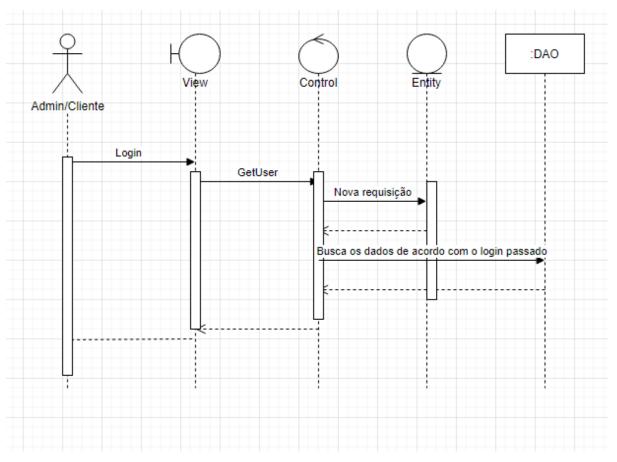
4 - MODELO DE INTERAÇÕES

O modelo de interações serve para demonstrar a nível arquitetônico as interações dentro do sistema entre o usuário e suas ações. No meu sistema, irei usar um ator representando um ator primário (ADMIN/CLIENTE) ou o ator referente ao cliente ou ao admin.

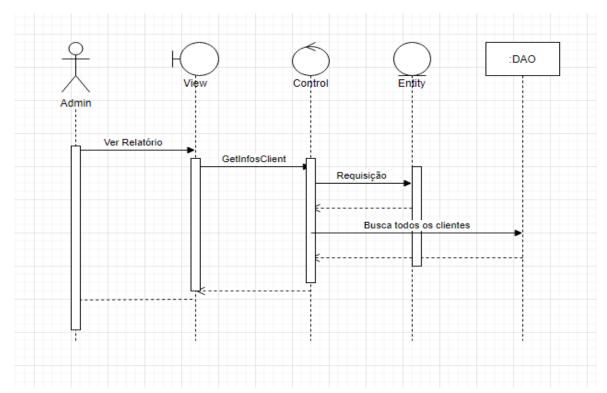
4.1. DIAGRAMA DE SEQUÊNCIA



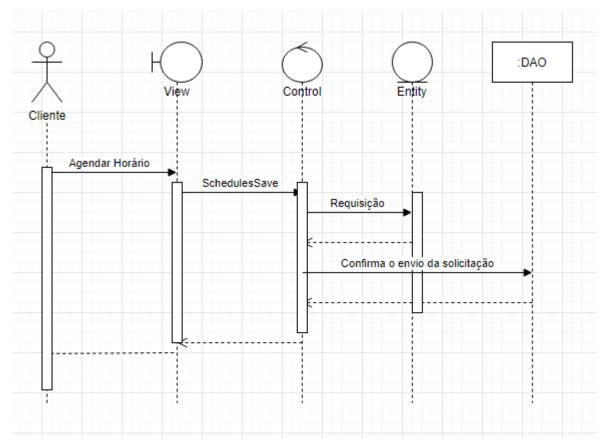
CSU01 - REGISTRAR



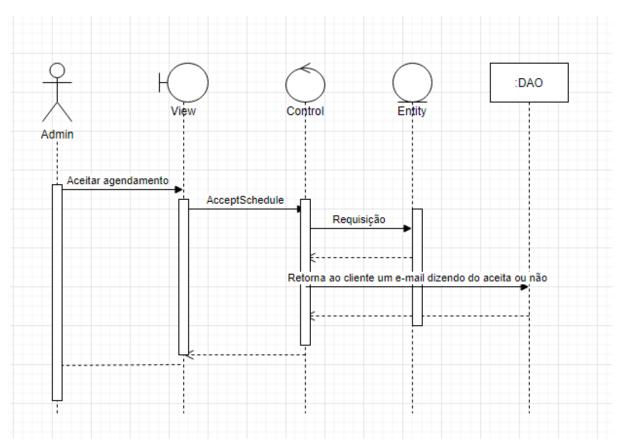
CSU02 - LOGIN



CSU03 - RELATÓRIOS



CSU04 – AGENDAR HORÁRIO



CSU05 – ACEITAR/RECUSAR SOLICITAÇÃO DO AGENDAMENTO

5 - MODELO DE TRANSIÇÃO DE ESTADOS

A seguir, iremos ao capítulo do modelo de transição de estados que basicamente irá demonstrar através de diagramas como se comporta cada estado de um objeto após transições de funcionalidades.

5.1. DIAGRAMA DE TRANSIÇÃO DE ESTADO

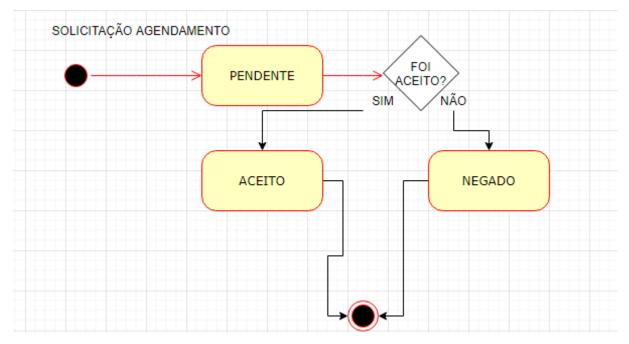


DIAGRAMA ACIMA REFERE-SE AO OBJETO AGENDAMENTO.

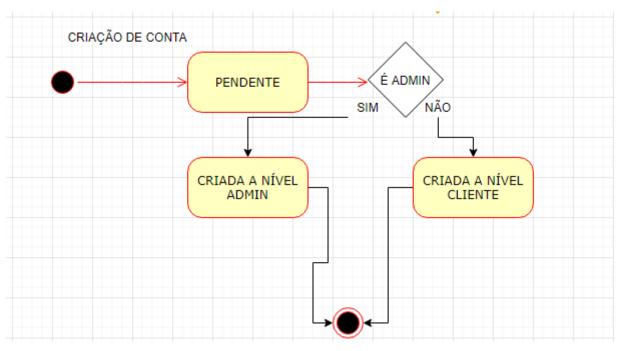


DIAGRAMA ACIMA REFERE-SE AO OBJETO CONTA.

6 - PROJETO DE BANCO DE DADOS

Um principio básico da T.I é que todo dado processado se torna uma informação; em um sistema há momentos que se é necessário trabalhar com salvamentos de dados para reutiliza-lo de alguma maneira, esse dado em base é salvo em um banco de dados, como modelamos esse banco de dados é o que tornará nosso dado uma informação,

portanto, um conjunto de caracteres que entrará na coluna nome na tabela cliente será a partir daquele momento uma informação, no caso, o nome do cliente. No nosso projeto também tivemos que utilizar um banco de dados e modelamos ele conforme nossa regra de negócio, tendo em mente sempre em deixa-lo de forma escalonável para futuras atualizações.

O Banco de dados utilizado foi o Sql Server, da Microsoft. No início quando pensamos na linguagem que iríamos desenvolver o nosso back-end, a primeira que veio em mente foi o C#, trabalhando com .NET CORE, logo, sendo o C# que é fortemente tipada e utiliza-se muito do OO, resolvi alinhar a base com o desenvolvimento e irmos para um banco relacional que aguentasse alta carga de dados e nos desse suporte para o desenvolvimento em .NET CORE.

6.1. PROJETO LÓGICO DE BANCO DE DADOS

[dbo].[Client]([ID_CLIENTE] [numeric](19, 0) IDENTITY(1,1) NOT NULL, [NAME_CLIENT] [varchar](50) NOT NULL,[DATA_CLIENT] [datetime] NOT NULL, [ID_CONTROL] [numeric](19, 0) NOT NULL, [USER_ID] [numeric](19, 0) NOT NULL)

[dbo].[User]([USER_ID] [numeric](19, 0) IDENTITY(1,1) NOT NULL,[USER] [varchar](max) NOT NULL,[PASSWORD] [varchar](max) NOT NULL, [DATA_CREATE] [datetime] NOT NULL,[TOKEN] [varchar](max) NOT NULL,[ID_CONTROL] [numeric](19, 0) NOT NULL,[TYPE_USER] [tinyint] NOT NULL, [USER_EMAIL] [varchar](max) NOT NULL)

[dbo].[ControlAcess]([ID_CONTROL] [numeric](19, 0) IDENTITY(1,1) NOT NULL, [NAME_CONTROL] [varchar](max) NOT NULl)

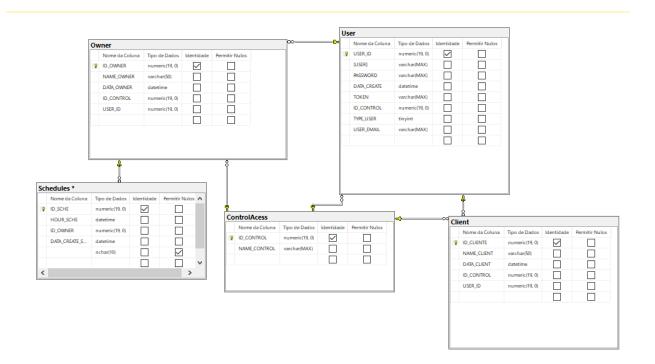
[dbo].[Owner]([ID_OWNER] [numeric](19, 0) IDENTITY(1,1) NOT NULL, [NAME_OWNER] [varchar](50) NOT NULL,[DATA_OWNER] [datetime] NOT NULL,[ID_CONTROL] [numeric](19, 0) NOT NULL, [USER_ID] [numeric](19, 0) NOT NULL)

[dbo].[Schedules]([ID_SCHE] [numeric](19, 0) IDENTITY(1,1) NOT NULL, [HOUR_SCHE] [datetime] NOT NULL, [ID_OWNER] [numeric](19, 0) NOT NULL, [DATA_CREATE_SCHE] [datetime] NOT NULL)

6.2. PROJETO FÍSICO DE BANCO DE DADOS

Agora iremos mostrar as tabelas de modo esquemático, com suas especificações e características, além dos scripts de criação de todo o banco de dados.

6.2.1. Tipificação dos atributos, domínio, obrigatoriedade e unicidade.



6.2.2. Script da base de dados

```
USE [Azul]
 GO
 SET QUOTED_IDENTIFIER ON
ECREATE TABLE [dbo].[Client](

[ID_CLIENTE] [numeric](19, 0) IDENTITY(1,1) NOT NULL,

[NAME_CLIENT] [varchar](50) NOT NULL,

[ID_CONTROL] [numeric](19, 0) NOT NULL,

[USER_ID] [numeric](19, 0) NOT NULL,
 [ID_CLIENTE] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
GO
GO
ALTER TABLE [dbo].[Client] WITH CHECK ADD CONSTRAINT [FK_ID_CONTROL] FOREIGN KEY([ID_CONTROL]) REFERENCES [dbo].[ControlAcess] ([ID_CONTROL]) GO
 ALTER TABLE [dbo].[Client] CHECK CONSTRAINT [FK_ID_CONTROL] GO
DALTER TABLE [dbo].[Client] WITH CHECK ADD CONSTRAINT [FK_USER_CLIENT] FOREIGN KEY([USER_ID])
REFERENCES [dbo].[User] ([USER_ID])
60
 ALTER TABLE [dbo].[Client] CHECK CONSTRAINT [FK_USER_CLIENT]
   USE [Azul]
GO
   /****** Object: Table [dbo].[Owner] Script Date: 21/06/2021 21:07:07 ******/
SET ANSI_NULLS ON
GO
    SET QUOTED_IDENTIFIER ON
 ECREATE TABLE [dbo].[Owner](

[ID OWNER] [numeric](19, 0) IDENTITY(1,1) NOT NULL,

[MAME (OWNER] [warchar](50) NOT NULL,

[DATA_OWNER] [datetime] NOT NULL,

[ID_CONTROL] [numeric](19, 0) NOT NULL,

[USER_ID] [numeric](19, 0) NOT NULL,

PRIMARY KEY CLUSTERED
   [ID_OWNER] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
ON [PRIMARY]
 □ALTER TABLE [dbo].[Owner] WITH CHECK ADD CONSTRAINT [FK_ID_CONTROL_OWNER] FOREIGN KEY([ID_CONTROL]) 

[REFERENCES [dbo].[ControlAcess] ([ID_CONTROL])
    ALTER TABLE [dbo].[Owner] CHECK CONSTRAINT [FK_ID_CONTROL_OWNER]
  ENALTER TABLE [dbo].[Owner] WITH CHECK ADD CONSTRAINT [FK_USER_OWNER] FOREIGN KEY([USER_ID]) [REFERENCES [dbo].[User] ([USER_ID]) GO
   ALTER TABLE [dbo].[Owner] CHECK CONSTRAINT [FK_USER_OWNER]
 USE [Azul]
GO
  /****** Object: Table [dbo].[ControlAcess] Script Date: 21/06/2021 21:06:41 ******/
  SET ANSI_NULLS ON
  SET QUOTED_IDENTIFIER ON
ECREATE TABLE [dbo],[ControlAcess](

[ID_CONTROL] [numeric](19, 0) IDENTITY(1,1) NOT NULL,

[NAME_CONTROL] [varchar](max) NOT NULL,

PRIMARY KEY CLUSTERED
  [ID_CONTROL] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

60
```

```
DEST [ADDITION OF SET TABLE [ADDITION OF THE CONTRAINT [FK_ID_OMER] FOREIGN KEY(ID_OMER])

SET MANUFACTURE [ADDITION OF THE CONTRAINT [FK_ID_OMER] FOREIGN KEY(ID_OMER])

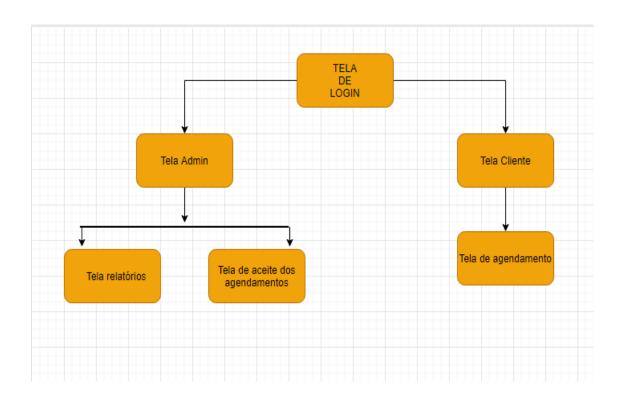
SET MANUFACTURE [ADDITION OF THE CONTRAINT [ADDITION OF THE CONTRAI
```

7 - PROJETO DE INTERFACE GRÁFICA

O projeto de uma interface é iniciado pela pesquisa, definindo os objetivos de acordo com o feedback de usuários e dados do mercado. (Fernando Lins, maio de 2020)

A Usabilidade é um dos fundamentos principais que compõe a experiência do usuário (UI), logo, assimilando a ideia de um sistema que tenha uma facilidade de ser usado para usuários leigos, demos o foco necessário para o sistema ter uma interface simples e que faça o que o sistema propõe.

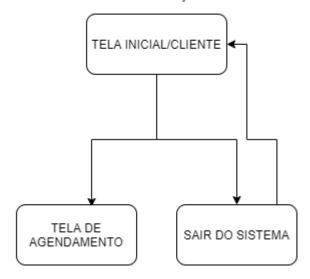
7.1. HIERARQUIA DAS TELAS.



7.2. MAPA DE NAVEGAÇÃO

MAPA DE NAVEGAÇÃO ADMIN TELA INICIAL/ADMIN SOLICITAÇÕES DE AGENDAMENTOS VER RELATÓRIOS SAIR DO SISTEMA

MAPA DE NAVEGAÇÃO CLIENTE



7.3. PADRONIZAÇÃO DE BOTÕES, ÍCONES E OUTROS ATALHOS

Botão logar: estará na tela de login, para quando o cliente quiser entrar dentro do sistema.



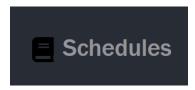
Botão Registre-se: estará na tela de LOGIN, o botão servirá para você ser redirecionado para tela de registro.



Botão Registrar: estará na tela de registro, irá servir para enviar a requisição para API para que possa ter a persistência dos dados na base.



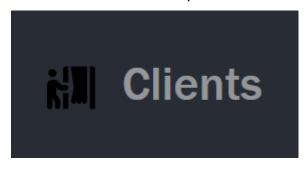
Botão Schedules: Aparecerá para o admin e irá redirecionar para a página de atendimento pendente.



Voltar: Botão para voltar para tela inicial, localizado na página de registro.



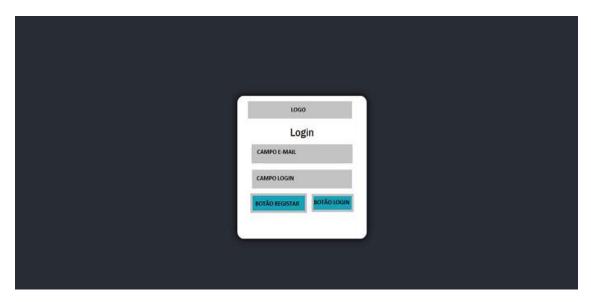
Clients: esse botão será para o admin verificar o relatório de clientes.



Out: aparecerá para ambos os perfis, será para sair do sistema.



7.4. LAYOUT DAS TELAS E RELATÓRIOS.



Tela de Login



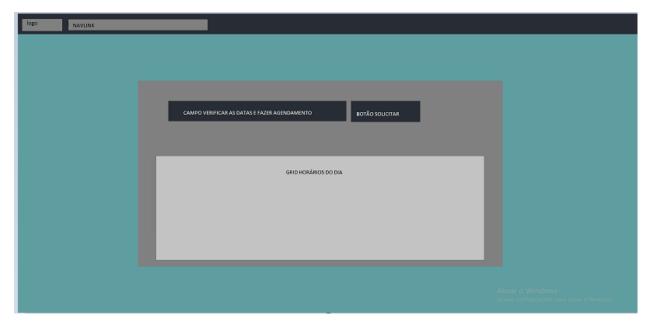
Tela Inicial Admin



Tela de Registro



Tela de aceite ou não



Tela agendamento cliente

8 - PROJETO DE IMPLANTAÇÃO

O projeto de implantação será necessário para alinhar as etapas de execução dele, sua instalação, monitoramento e o que será necessário para implementá-lo dentro de uma estrutura organizacional. Essa etapa é importante para documentar o modo de como será implantado alguns processos e seus procedimentos necessários. Para que a implantação seja bem sucedida, é necessário planejar. (João Pimenta, maio de 2021).

8.1. PROCEDIMENTOS DE IMPLANTAÇÃO

Para iniciar os procedimentos de implantação do sistema, será necessário:

- Dados da empresa para poder fazer a criação da conta de admin:
- Para podermos dar início no sistema, temos que configurá-lo para cada empresa.
 - Backup:
- O Backup é feito pela empresa responsável pelo armazenamento em

 Nuvem

8.2. AMBIENTE OPERACIONAL NECESSÁRIO E A SEQUÊNCIA DE MÓDULOS A SEREM IMPLANTADOS.

A única necessidade será conexão com a Internet, pois o sistema é totalmente dependente de Internet. Questões como dados em Nuvem, backup, acessos também em Nuvem, etc, necessitam 100% da Internet, portanto, será mais do que necessário a conexão com a rede.

8.3. ESTRATÉGIA DE IMPLEMENTAÇÃO

A nível cliente, será disponibilizado os seguintes módulos:

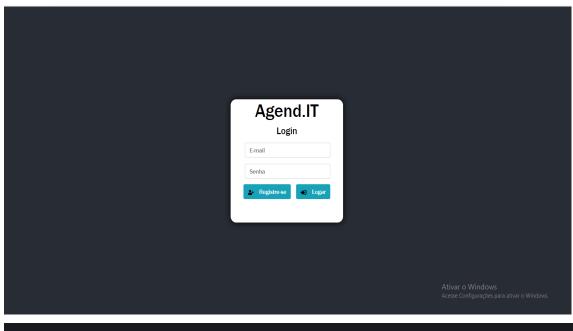
 Agendamento de datas e posteriormente, também será implementado histórico de agendamentos (solicitações e agendamentos concluídos);

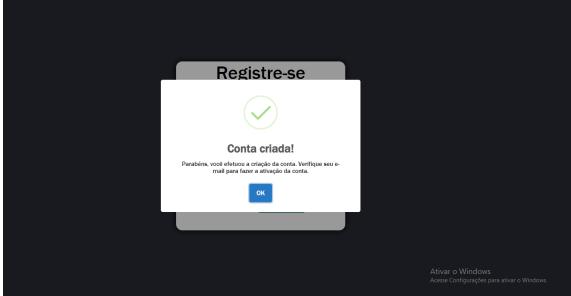
A nível admin, de início, será disponibilizado os seguintes módulos:

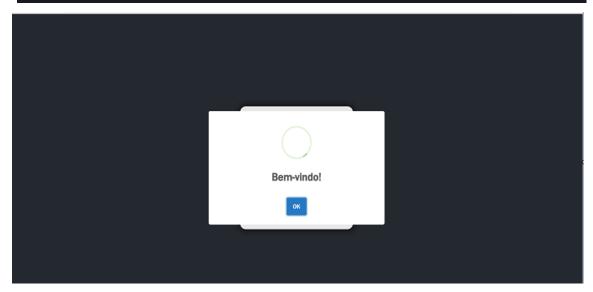
- Relatório de cliente;
- Módulo de solicitações em aberto;

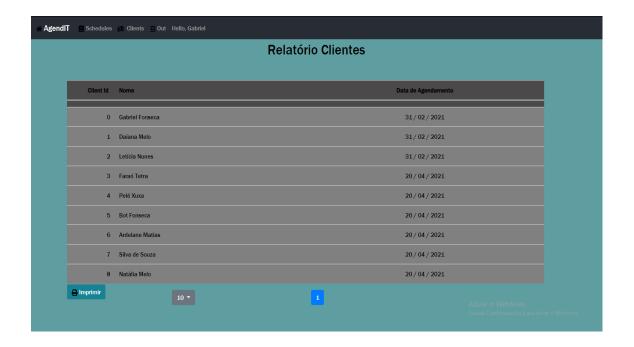
9 - AMOSTRA DO SISTEMA

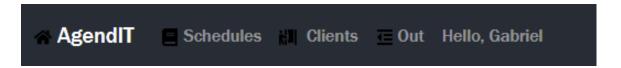
O sistema conta com um sistema de login padrão: e-mail, senha e nome. Validações de dados antes de logar, etc. Basicamente, o sistema é baseado em agendamentos/locações de horários, além de contar com relatórios (até o momento, somente o de cliente foi implantado), também conta com um backend escrito visando dar escalabilidade para futuras features. O sistema funciona da seguinte maneira a nível cliente: você cria uma conta e faz seu login, na tela inicial aparece a tela para você efetuar a solicitação de agendamento; nesse momento já é disparado um e-mail para a loja (cadastrado no nível admin) sobre o agendamento. Para esse agendamento ser concluído é necessário que o admin logue no sistema e aceite ou negue, logo após a resposta é disparado um e-mail ao cliente (cadastrado no cadastro da conta cliente) informando sobre a resposta do agendamento.











10 - CONCLUSÕES E TRABALHOS FUTUROS

Esse foi um sistema fácil de desenvolver, porém, difícil de modelar. Os inúmeros relacionamentos necessários baseados nos requisitos do sistema fizeram com que o desenvolvimento fosse um pouco mais demorado do esperado, porém a forma como foi modelado e os inúmeros diagramas feitos deram uma maior facilidade na hora de transcrever para o código. Entretanto, há diversas features que possam ser aplicadas em um futuro, como por exemplo: desenvolver mais relatórios, relatórios financeiros, agendamentos mensais, inclusões de filtros, entre outros, além de chat online cliente - admin, suporte, entre outros.

Este projeto teve como objetivo automatizar a entrega de um agendamento com um menor custo e uma maior flexibilidade.

11 - BIBLIOGRAFIA

Teixeira, Danielle. Requisitos de software, sua importância. 2018. Disponível em https://medium.com/lfdev-blog/como-escrever-requisitos-de-software-de-forma-simples-e-garantir-o-mínimo-de-erros-no-sistema-app-74df2ee241cc. Acessado em junho de 2021.

Sebrae Alagoas. Processo de negócios na sua empresa. 2020. Disponível em https://blog.sebraealagoas.com.br/gestao/bpm-o-guia-completo-para-voce-aplicar-a-modelagem-de-processos-de-negocio-na-sua-empresa/. Acessado em junho de 2021.

Verissímo, Ricardo. Levantando requisitos para uma empresa. 2007. Disponível em

https://www.baguete.com.br/artigos/296/ricardo-verissimo/05/11/2007/levantamento-de-requisitos-e-mapeamento-de-processos. Acessado em junho de 2021.

Ventura Plínio. O caso de uso. 2016.

Disponível em https://www.ateomomento.com.br/o-que-e-caso-de-uso/.

Acessado em junho de 2021.

Lins, Fernando. A importância de uma Interface Gráfica. 2020. Disponível em https://medium.com/@fernando1lins/analisando-e-redesenhando-a-interface-gráfica-do-playstation-4-parte-3-e8914db882f3. Acessado em junho de 2021.

Pimenta, João. Descrevendo uma implantação de software. 2021. Disponível em https://www.nomus.com.br/blog-industrial/para-que-serve-um-plano-de-projeto-de-implantação-de-um-software-erp/. Acessado em junho de 2021.

Barcelos Carlos, Diagrama de Classes. 2019. Disponível em https://medium.com/documentaçao-uml/diagrama-de-classes-ba91a9d29575. Acessado em junho de 2021.