



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MATO
GROSSO DO SUL
CAMPUS NAVIRAÍ
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

LUIS GUSTAVO GOIS CHIQUITIN

REXTON-OS - GERENCIAMENTO DE ORDEM DE SERVIÇOS

NAVIRAÍ

2023

LUIS GUSTAVO GOIS CHIQUITIN

REXTON-OS - GERENCIAMENTO DE ORDEM DE SERVIÇOS

Trabalho de conclusão de curso apresentado a banca examinadora como requisito do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Mato Grosso do Sul.

Orientador: Prof. M^e. ALISSON GASPAR CHIQUITTO

NAVIRAÍ

2023

Luis Gustavo Gois Chiquitin

Rexton-OS - Gerenciamento de Ordem de Serviços/ Luis Gustavo Gois Chiquitin. – Naviraí, 2023-

45 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. M^e. ALISSON GASPAR CHIQUITTO

Monografia – INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MATO GROSSO DO SUL
CAMPUS NAVIRAÍ

TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS, 2023.

1. Palavra-chave1. 2. Palavra-chave2. 2. Palavra-chave3. I. Orientador. II. Universidade xxx. III. Faculdade de xxx. IV. Título

LUIS GUSTAVO GOIS CHIQUITIN

REXTON-OS - GERENCIAMENTO DE ORDEM DE SERVIÇOS

Trabalho de conclusão de curso apresentado a banca examinadora como requisito do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Mato Grosso do Sul.

Aprovado pela banca examinadora em
BANCA EXAMINADORA:

Prof. M^e. ALISSON GASPAS CHIQUITTO
Instituto Federal de Educação, Ciência e Tecnologia do Mato
Grosso do Sul - IFMS

NAVIRAÍ
2023

RESUMO

A cada dia são criados mais prestadores de serviços no Brasil. Para essas empresas, é necessário utilizar um sistema de TI para gerenciar, melhorar e refinar os processos de gestão da empresa. Um módulo importante para esses sistemas é o que trata do gerenciamento de ordens de serviço. O objetivo do trabalho é apresentar o sistema de gestão de ordens de serviço para a empresa Rexton.

O sistema será criado para o ambiente mobile, onde é possível realizar esse gerenciamento utilizando a plataforma Android com o framework React-native e o banco de dados PostgreSQL. Como resultado, será criado um sistema que permite gerenciar ordens de serviço, controlar estoques de peças e clientes.

Palavras-chave: Javascript, desenvolvimento, banco de dados, postgresSQL, ordens de serviço.

ABSTRACT

Every day more services are produced in Brazil. For these companies, it is necessary to use an IT system to manage, improve and refine the processes of company management. An important module for these systems is the one that deals with the service order management. The objective of this work is to present the system of management of service requests for and company Rexton.

The system will be created for the mobile environment, where it is possible to carry out this management using the Android platform with the React-native framework and the database PostgreSQL. As a result, a system will be created that allows managing orders of service, control stocks of parts and customers.

Keywords: Javascript, development, database, postgresSQL, orders service.

LISTA DE ILUSTRAÇÕES

Figura 1 – Diagrama caso de uso - Efetuar Login	14
Figura 2 – Diagrama caso de uso - Cadastrar Usuário	15
Figura 3 – Diagrama caso de uso - Cadastrar Funcionário	16
Figura 4 – Diagrama caso de uso - Cadastrar Cliente	17
Figura 5 – Diagrama caso de uso - Cadastrar Serviços	18
Figura 6 – Diagrama caso de uso - Cadastrar Ordem de Serviço	19
Figura 7 – Diagrama de Atividades - Visão Geral do Sistema	20
Figura 8 – Diagrama de Atividades - Cadastro do Sistema	21
Figura 9 – Diagrama de Atividades - Cadastro do Sistema	21
Figura 10 – Diagrama de Atividades - Cadastro do Sistema	22
Figura 11 – Diagrama de Atividades - Cadastro do Sistema	22
Figura 12 – Diagrama de Atividades - Cadastro do Sistema	23
Figura 13 – Diagrama de Atividades - Cadastro do Sistema	23
Figura 14 – Diagrama de Entidade Relacionamento	24
Figura 15 – Diagrama de Classes	25
Figura 16 – Tela de Login	29
Figura 17 – Cadastrar USUÁRIOS	30
Figura 18 – Listagem de OS	31
Figura 19 – Cadastrar ORDEM DE SERVIÇO	32
Figura 20 – Cadastrar Clientes	33
Figura 21 – Cadastrar Funcionário	34
Figura 22 – Cadastrar Tipos de Serviços	35

LISTA DE TABELAS

Tabela 1 – Caso de uso - Efetuar Login	14
Tabela 2 – Caso de uso - Cadastrar Usuário	15
Tabela 3 – Caso de uso - Cadastrar Funcionário	16
Tabela 4 – Caso de uso - Cadastrar Cliente	17
Tabela 5 – Caso de uso - Cadastrar Serviços	18
Tabela 6 – Caso de uso - Cadastrar Ordem de Serviço	19

SUMÁRIO

	INTRODUÇÃO	10
1	FUNDAMENTAÇÃO TEÓRICA	11
1.1	PRESTAÇÃO DE SERVIÇOS	11
1.2	ORDEM DE SERVIÇO	11
1.3	SISTEMAS DE INFORMAÇÃO	11
2	JUSTIFICATIVA	12
3	METODOLOGIA	13
3.1	CASOS DE USO	13
3.1.1	CASOS DE USO: EFETUAR LOGIN	13
3.1.2	CASOS DE USO: CADASTRAR USUÁRIO	14
3.1.3	CASOS DE USO: CADASTRAR FUNCIONÁRIO	15
3.1.4	CASOS DE USO: CADASTRAR CLIENTE	16
3.1.5	CASOS DE USO: CADASTRAR SERVIÇOS	17
3.1.6	CASOS DE USO: CADASTRAR ORDEM DE SERVIÇO	18
3.2	DIAGRAMA DE ATIVIDADES	19
3.2.1	VISÃO GERAL	19
3.2.2	CADASTROS	20
3.2.2.1	CADASTRAR USUÁRIO	20
3.2.2.2	CADASTRAR FUNCIONÁRIO	21
3.2.2.3	CADASTRAR SERVIÇOS	21
3.2.2.4	CADASTRAR CLIENTE	22
3.2.2.5	CADASTRAR ORDEM DE SERVIÇO	22
3.3	DIAGRAMA DE ENTIDADE RELACIONAMENTO	23
3.4	DIAGRAMA DE CLASSES	24
3.5	DICIONÁRIO DE DADOS	25
3.5.1	ENTIDADE DE CLIENTES	26
3.5.2	ENTIDADE DE FUNCIONÁRIOS	26
3.5.3	ENTIDADE DE ORDEM DE SERVIÇO	27
3.5.4	ENTIDADE DE SERVIÇOS	27
3.5.5	ENTIDADE DE ORDEM USUÁRIOS	28
3.6	TELAS DO SISTEMA	29
3.6.1	TELA DE LOGIN	29
3.6.2	TELA CADASTRAR USUÁRIOS	30

3.6.3	LISTAGEM DE ORDEM DE SERVIÇO	31
3.6.4	TELA CADASTRAR ORDEM DE SERVIÇO	32
3.6.5	TELA CADASTRAR CLIENTES	33
3.6.6	TELA CADASTRAR FUNCIONÁRIO	34
3.6.7	TELA CADASTRAR TIPOS DE SERVIÇOS	35
3.7	LEVANTAMENTO DE INFORMAÇÕES	36
3.8	LEVANTAMENTO DE REQUISITOS	36
3.8.1	ENTREVISTA REALIZADA NA EMPRESA	37
4	OBJETIVOS	38
4.1	Específicos	38
5	METODOLOGIA	39
5.1	React-Native	39
5.2	Node.JS	39
5.3	PostgreSQL	40
5.4	TypeORM	40
5.5	Astah	41
5.6	Visual Studio Code	42
6	CONCLUSÃO	43
	REFERÊNCIAS	44

INTRODUÇÃO

O trabalho tem como objetivo uma empresa do setor de terraplenagem chamada Rexton, que possui uma riqueza de informações e possui apenas um departamento para gerenciar, embora seja necessário criar uma ferramenta para gerenciar os serviços prestados. Da necessidade de organizar e controlar a quantidade de informações, surge a necessidade de construir um sistema eletrônico de gestão de ordens de serviço.

Atualmente, um grande número de serviços é gerado no dia a dia da empresa, portanto, é necessário criar relatórios contendo fluxo de caixa, histórico de ordens de serviço, podendo separar por status, como pendentes, cancelados e concluídos, para exemplo, além de controlar os principais equipamentos da empresa. Essas informações devem ser armazenadas por um período de tempo predeterminado.

Atualmente, os relatórios da empresa são feitos manualmente e armazenados em pastas e alocados em espaço físico, e muitas vezes acaba não sendo o caminho certo a seguir. A inconveniência será enorme ao eventualmente pesquisar ou pesquisar qualquer um desses registros, pois muitas vezes o arquivo acumulado acaba se perdendo, ou sua localização é demorada e ineficiente porque você tem que ir página por página.

A empresa aqui mencionada pode ser considerada como MEI (Microempreendedor), e possui somente um funcionário. Diante disso, a principal questão a ser discutida é: como controlar efetivamente todos os serviços e execuções?

O sistema de ordens de serviço foi projetado para ajudar a administração a lidar com os serviços prestados dentro da empresa, uma questão que afeta totalmente a Rexton.

1 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os fundamentos teóricos dos principais tópicos abordados neste trabalho, como prestação de serviços, ordem de serviço e sistemas de informação.

1.1 PRESTAÇÃO DE SERVIÇOS

O [SENAC \(2001\)](#) esclarece: "Um serviço é o resultado de pelo menos uma atividade realizada na interface do fornecedor com o cliente. Geralmente é intangível". De acordo com [Nitahara \(2020\)](#), a prestação de serviços representa 70% das empresas do país e é responsável por cerca de 30% do PIB e em 2019, o setor foi líder na geração de empregos, com mais de 382 mil vagas formais abertas, contra 145 mil do comércio e 71 mil da construção civil. A qualidade dos serviços prestados é da maior importância, pois cada vez mais clientes procuram serviços e esperam sempre a qualidade desses serviços. Para atingir a qualidade do serviço, é necessário criar um ambiente especial dentro da empresa para prestar um serviço de excelência aos clientes. é o compromisso de todos os membros da organização. ([SENAC, 2001](#))

1.2 ORDEM DE SERVIÇO

De acordo com SAP (Systems Applications Products) ([SILVA, 2020](#)), uma ordem de serviço é um contrato de curto prazo entre um provedor de serviço e um destinatário de serviço no qual um único serviço é especificado no pedido e o faturamento relacionado ao recurso é realizado. É uma solicitação para realizar uma atividade de serviço em um objeto de manutenção de uma empresa cliente em uma data específica. Além disso, de acordo com a SAP, as ordens de serviço são utilizadas como uma utilidade para monitorar as atividades de um determinado serviço prestado.

1.3 SISTEMAS DE INFORMAÇÃO

Segundo [Rezende e Abreu \(2008\)](#), sistemas de informação são todos os sistemas que geram informações para realizar ações e auxiliar a tomada de decisões. É comum as empresas automatizarem processos por meio de um ou mais sistemas de informação.

Segundo Rezende e Abreu, toda empresa moderna preocupada com sua sustentabilidade e competitividade deve também focar na execução e organização das atividades de planejamento estratégico, sistemas de informação e gestão de TI.

2 JUSTIFICATIVA

O motivo da implementação de um sistema de ordens de serviço é a necessidade de consolidar todas as informações sobre os serviços prestados e equipamentos pertencentes à empresa. O grande problema que foi resolvido é a questão de poder saber qual o próximo serviço a ser feito, cadastrar clientes, e poder atualizar o preço dos serviços prestados com mais facilidade, tendo em vista que o aplicativo será para dispositivos mobile.

Os relatórios armazenados podem ser continuamente necessários em diferentes situações para verificação do serviço, verificação do status do serviço, execução, autorização de consultas externas. Dessa forma, os relatórios que tratam do dia a dia da empresa estarão sempre bem organizados e dispostos para consultas de forma rápida e estratégica.

3 METODOLOGIA

Neste capítulo serão apresentados o que e como foi feito todo o desenvolvimento da aplicação. Antes de qualquer coisa devemos primeiramente entender o significado de metodologia, que nada mais é onde é permitido descrever o caminho percorrido para atingir o objetivo proposto, buscando solucionar a questão de pesquisa.

Primeiramente foi feito totalmente do zero um serviço de API no backend para realizar todas as regras de negocio da aplicação, como cadastrar as informações, listar, editar e excluir.

Para a criação da API foi utilizada a linguagem javascript em conjunto com o NodeJS, o framework TypeORM e o gerenciador de banco de dados, PostgreSQL.

O NodeJS foi utilizado no desenvolvimento das rotas para serem acessadas pelo frontend que é a aplicação em si.

Já o TypeORM é nada mais que uma técnica que aproxima o paradigma de desenvolvimento de aplicativos orientados a objetos do paradigma de banco de dados relacional. A justificativa para se ter utilizado esse framework é evitar um problema chamado impedância objeto-relacional, logo o typeorm define uma técnica para realizar a conciliação onde ele define o modo como os dados serão mapeados entre os ambientes, como serão acessados e gravados. Isso diminui o tempo de desenvolvimento, uma vez que não é necessário perder tempo fazendo essa configuração.

Por fim temos o PostgreSQL que é nada mais que um gerenciador de banco de dados relacionados que otimiza muito o trabalho de quem precisa administrar informações nesses níveis.

A fim de expor detalhadamente os procedimentos e métodos que foram adotados no estudo, foi dividido em oito tópicos.

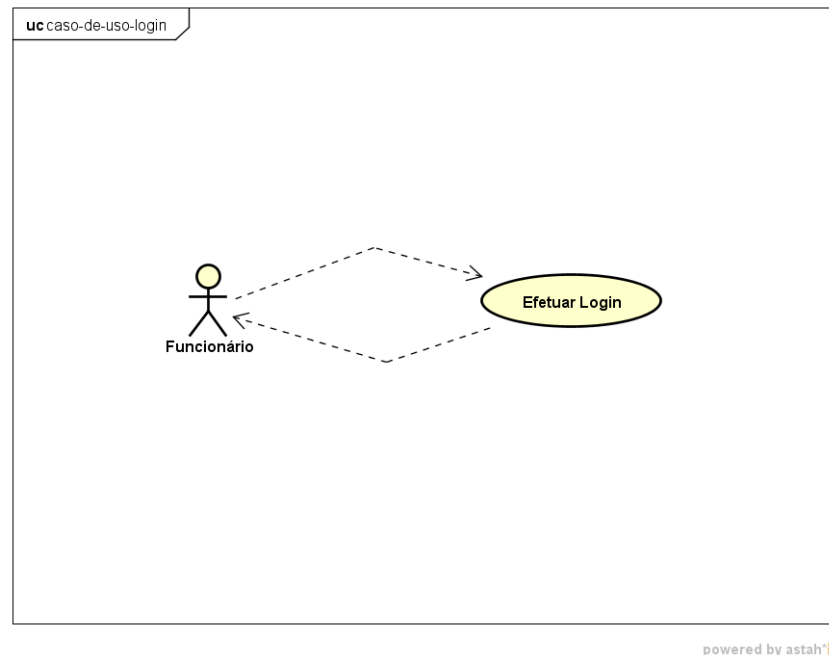
3.1 CASOS DE USO

3.1.1 CASOS DE USO: EFETUAR LOGIN

Ator: Funcionário

Pré-Requisito: Ter Cadastrado Senha e Login do Funcionário.

Figura 1 – Diagrama caso de uso - Efetuar Login



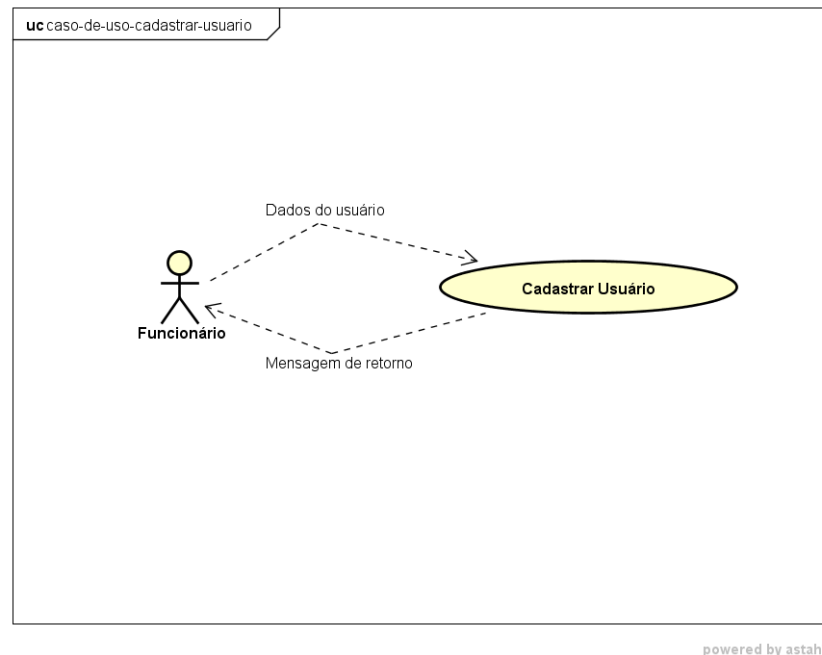
Ação do Ator	Resposta do Sistema ou Exceções
1-O Usuário inicia o Sistema de Ordem de Serviço.	2- O Sistema exibe a janela com dois campos para o Usuário informar Login e Senha.
3- O Usuário informa Login, Senha e Confirma.	4- O Sistema faz a validação do Login e Senha informada com os dados de Login e Senha cadastrado no Sistema, [Passo 7].
	5-O Sistema abre dando acesso a todo o Sistema com exceção do Funcionário que somente terá acesso a tela de geração de O.S.
6- O Usuário já pode utilizar O Sistema.	
7-(Exceção) O Usuário informa Login e Senha incorretas retorna ao passo 2.	

Tabela 1 – Caso de uso - Efetuar Login

3.1.2 CASOS DE USO: CADASTRAR USUÁRIO

Ator: Funcionário

Figura 2 – Diagrama caso de uso - Cadastrar Usuário



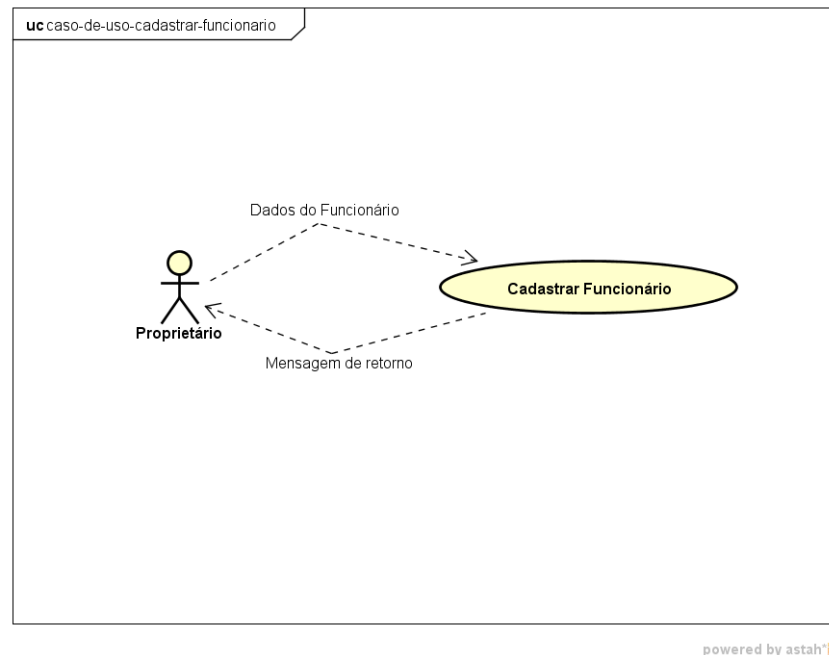
Ação do Ator	Resposta do Sistema ou Exceções
1- O Usuário inicia solicitando a tela de Login.	2- O Sistema inicia abrindo a tela de Login.
3- O Usuário informa seu Login e Senha ao Sistema.	4- O Sistema valida a Senha e Login e abre a tela inicial dando acesso a todo o Sistema, [passo 9].
5- O Usuário seleciona a tela de cadastro de Usuário do Menu.	6- O Sistema abre a tela de cadastro de Usuário.
7- O Usuário Proprietário entra com os dados do Usuário, Nome, RG, CPF, Endereço, E-mail, Telefone, Senha	8- O Sistema Salva e Retorna ao passo 6 .
9- (Exceção) O Usuário informa Login ou Senha Incorreta e o Sistema e retorna ao passo 2.	10- (Exceção) O Usuário Cancela o cadastro e retorna ao passo 6.
11- (Exceção) – O Sistema informa que o CPF e o RG informado já esta sendo utilizado e volta ao passo 7.	

Tabela 2 – Caso de uso - Cadastrar Usuário

3.1.3 CASOS DE USO: CADASTRAR FUNCIONÁRIO

Ator: Proprietário

Figura 3 – Diagrama caso de uso - Cadastrar Funcionário



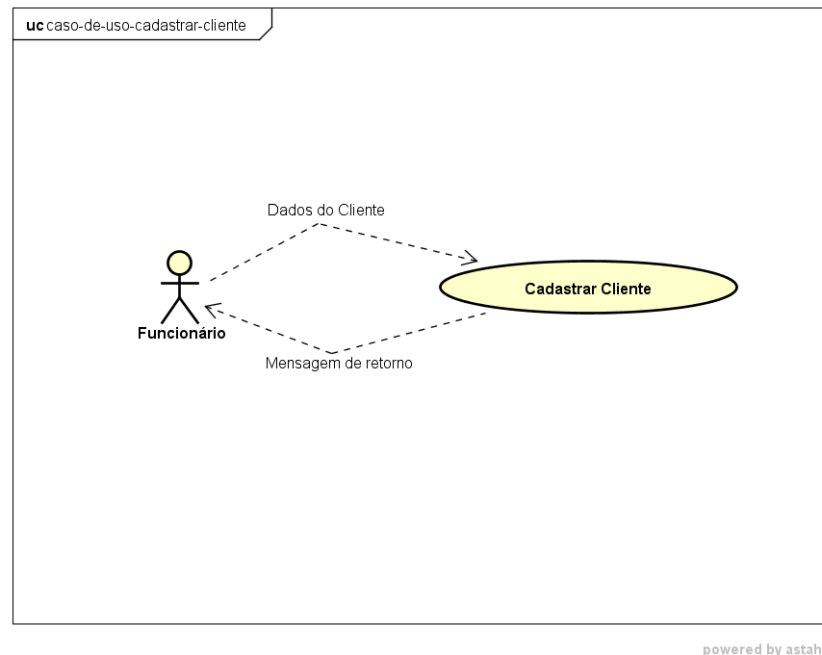
Ação do Ator	Resposta do Sistema ou Exceções
1- O Usuário inicia solicitando a tela de Login.	2- O Sistema inicia abrindo a tela de Login.
3- O Usuário informa seu Login e Senha ao Sistema.	4- O Sistema valida a Senha e Login e abre a tela inicial dando acesso a todo o Sistema, [passo 9].
5- O Usuário seleciona a tela de cadastro de Funcionário do Menu.	6- O Sistema abre a tela de cadastro de Funcionário.
7- O Usuário entra com os dados do Funcionário, Nome, CPF, Endereço, Data de nascimento, Data de Admissão.	8- O Sistema Salva e Retorna ao passo 6 .
9- (Exceção) O Usuário informa Login ou Senha Incorreta e o Sistema e Retorna ao passo 2.	10- (Exceção) O Usuário Cancela o cadastro e retorna ao passo 6.
11- (Exceção) – O Sistema informa que as Senhas não conferem e volta ao passo 7.	

Tabela 3 – Caso de uso - Cadastrar Funcionário

3.1.4 CASOS DE USO: CADASTRAR CLIENTE

Ator: Proprietário

Figura 4 – Diagrama caso de uso - Cadastrar Cliente



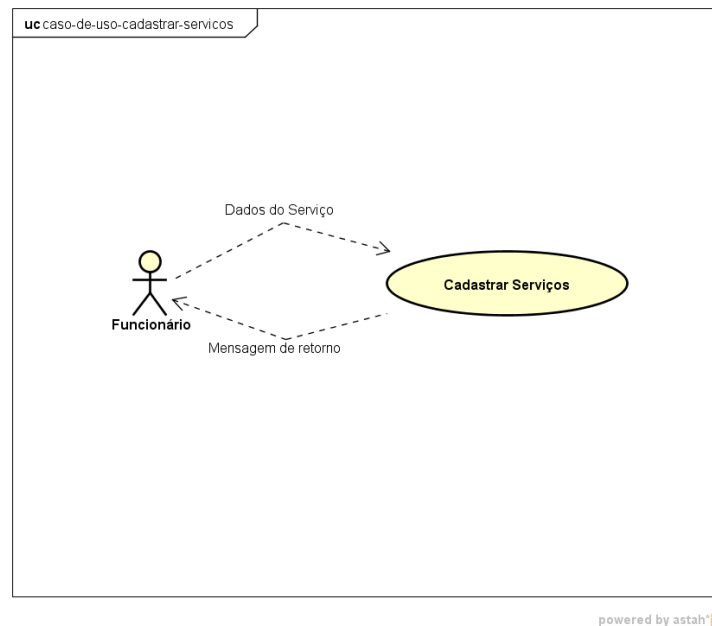
Ação do Ator	Resposta do Sistema ou Exceções
1- O Usuário inicia solicitando a tela de Login.	2- O Sistema inicia abrindo a tela de Login.
3- O Usuário informa seu Login e Senha ao Sistema.	4- O Sistema valida a Senha e Login e abre a tela inicial dando acesso a todo o Sistema, [passo 9].
5- O Usuário seleciona a tela de cadastro de Cliente do Menu.	6- O Sistema abre a tela de cadastro de Cliente.
7- O Usuário entra com os dados do Cliente, Nome, CPF, Endereço, Cidade, Estado, Telefone, E-mail e Data de Nascimento	8- O Sistema Salva e Retorna ao passo 6 .
9- (Exceção) O Usuário informa Login ou Senha Incorreta e o Sistema e Retorna ao passo 2.	10- (Exceção) O Usuário Cancela o cadastro e retorna ao passo 6.
11- (Exceção) – O Sistema informa que o CPF informado já esta sendo utilizado e volta ao passo 7.	

Tabela 4 – Caso de uso - Cadastrar Cliente

3.1.5 CASOS DE USO: CADASTRAR SERVIÇOS

Ator: Proprietário

Figura 5 – Diagrama caso de uso - Cadastrar Serviços



Ação do Ator	Resposta do Sistema ou Exceções
1- O Usuário inicia solicitando a tela de Login.	2- O Sistema inicia abrindo a tela de Login.
3- O Usuário informa seu Login e Senha ao Sistema.	4- O Sistema valida a Senha e Login e abre a tela inicial dando acesso a todo o Sistema, [passo 9].
5- O Usuário seleciona a tela de cadastro de Serviços do Menu.	6- O Sistema abre a tela de cadastro de Serviços.
7- O Usuário entra com os dados do Serviço, Descrição do serviço e valor do serviço.	8- O Sistema Salva e Retorna ao passo 6 .
9- (Exceção) O Usuário informa Login ou Senha Incorreta e o Sistema e retorna ao passo 2.	10- (Exceção) O Usuário Cancela o cadastro e retorna ao passo 6.

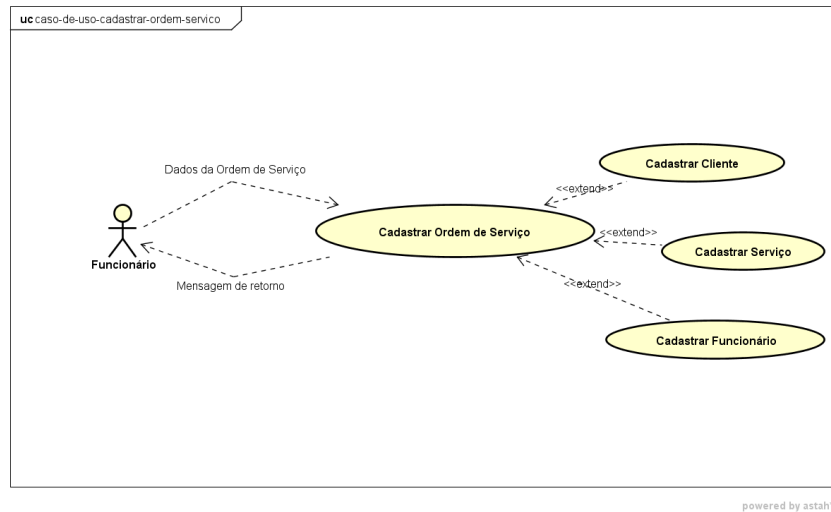
Tabela 5 – Caso de uso - Cadastrar Serviços

3.1.6 CASOS DE USO: CADASTRAR ORDEM DE SERVIÇO

Ator: Proprietário

Pré-Requisito: Ter cadastrado o cliente, funcionário e o serviço.

Figura 6 – Diagrama caso de uso - Cadastrar Ordem de Serviço



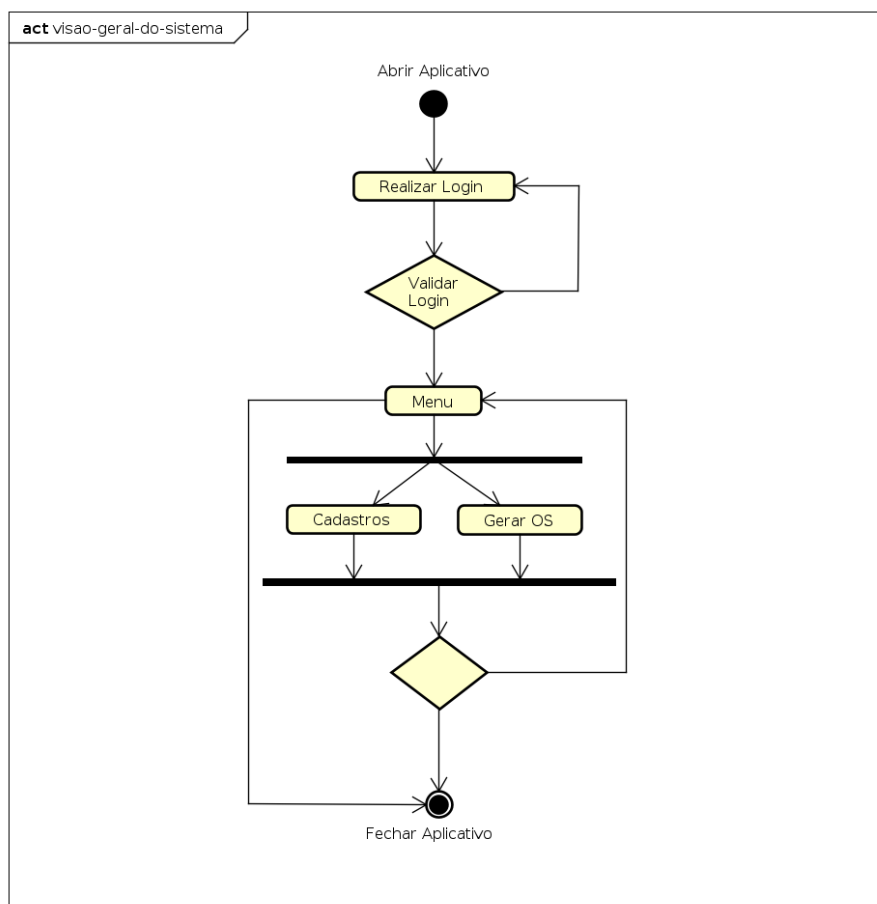
Ação do Ator	Resposta do Sistema ou Exceções
1- O Usuário inicia solicitando a tela de Login.	2- O Sistema inicia abrindo a tela de Login.
3- O Usuário informa seu Login e Senha ao Sistema.	4- O Sistema valida a Senha e Login e abre a tela inicial dando acesso a todo o Sistema, [passo 9].
5- O Usuário seleciona a tela de cadastro de Ordem de Serviço do Menu.	6- O Sistema abre a tela de cadastro de Ordem de Serviço.
7- O Usuário entra informa qual o cliente, funcionário e o serviço.	8- O Sistema Salva e Retorna ao passo 6 .
9- (Exceção) O Usuário informa Login ou Senha Incorreta e o Sistema e retorna ao passo 2.	10- (Exceção) O Usuário Cancela o cadastro e retorna ao passo 6.

Tabela 6 – Caso de uso - Cadastrar Ordem de Serviço

3.2 DIAGRAMA DE ATIVIDADES

3.2.1 VISÃO GERAL

Figura 7 – Diagrama de Atividades - Visão Geral do Sistema



3.2.2 CADASTROS

3.2.2.1 CADASTRAR USUÁRIO

Figura 8 – Diagrama de Atividades - Cadastro do Sistema

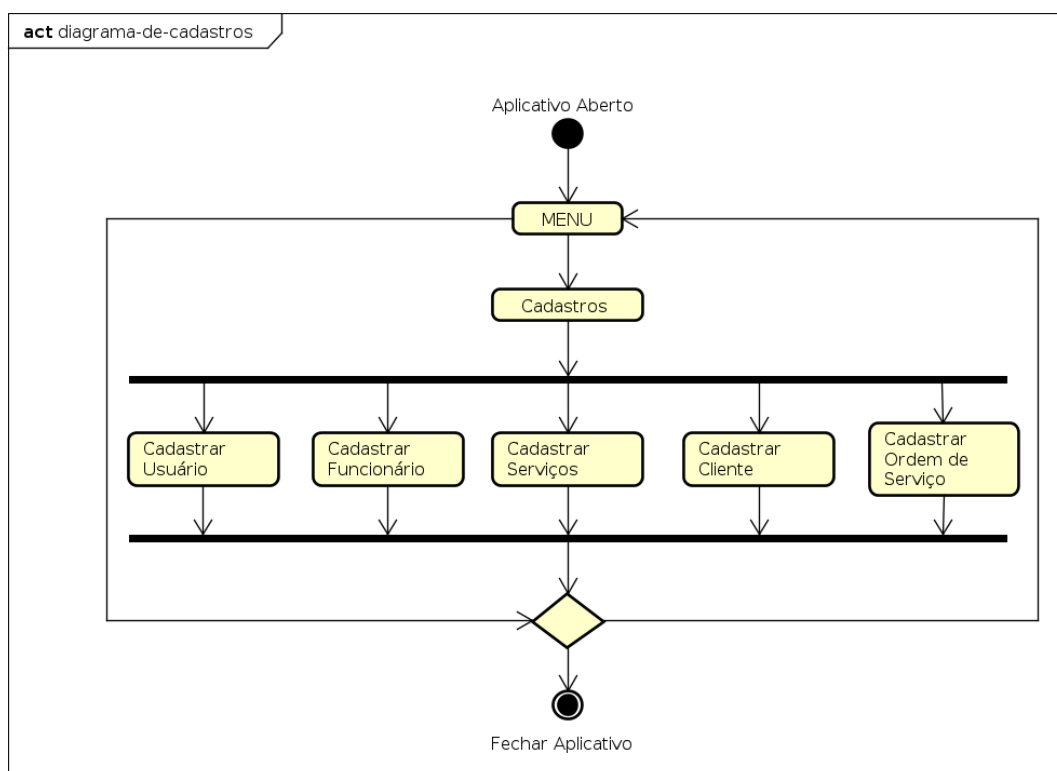
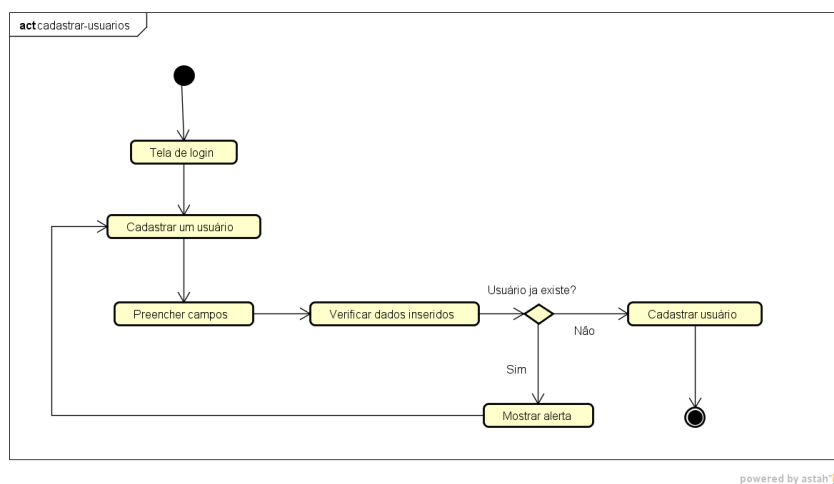


Figura 9 – Diagrama de Atividades - Cadastro do Sistema



3.2.2.2 CADASTRAR FUNCIONÁRIO

3.2.2.3 CADASTRAR SERVIÇOS

Figura 10 – Diagrama de Atividades - Cadastro do Sistema

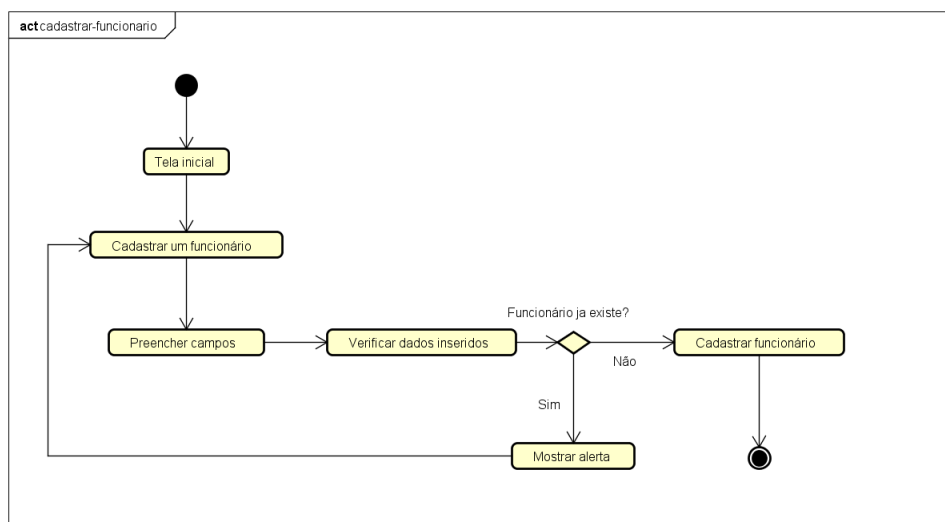
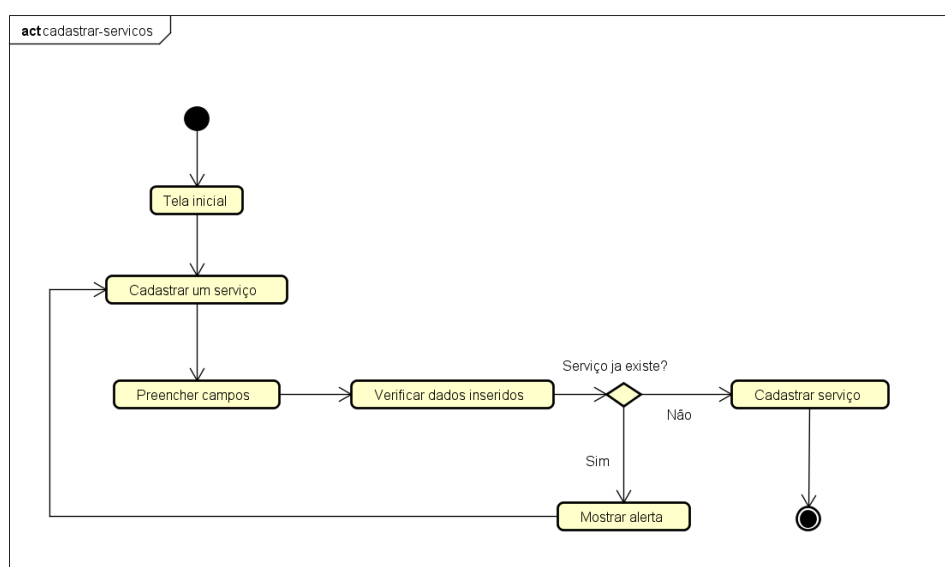


Figura 11 – Diagrama de Atividades - Cadastro do Sistema



3.2.2.4 CADASTRAR CLIENTE

3.2.2.5 CADASTRAR ORDEM DE SERVIÇO

Figura 12 – Diagrama de Atividades - Cadastro do Sistema

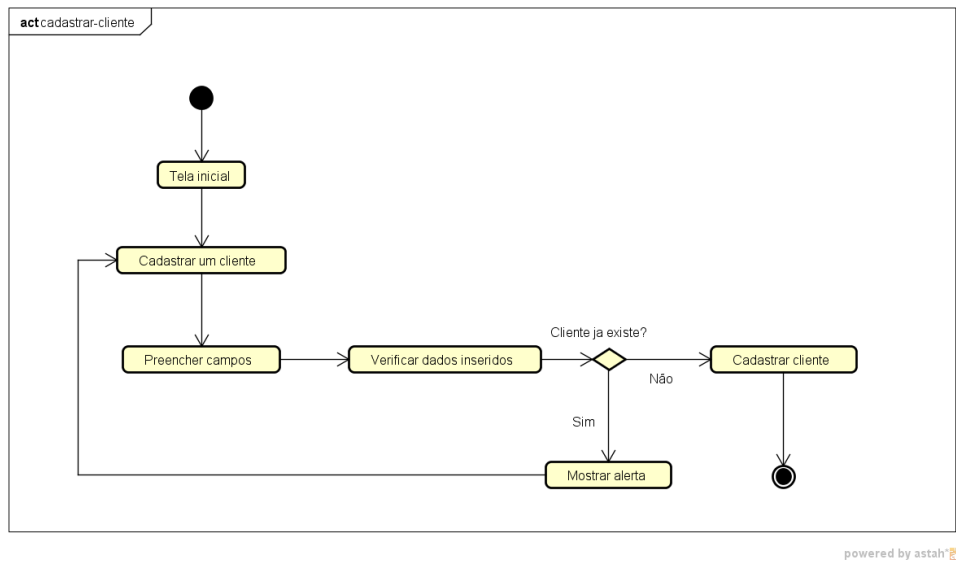
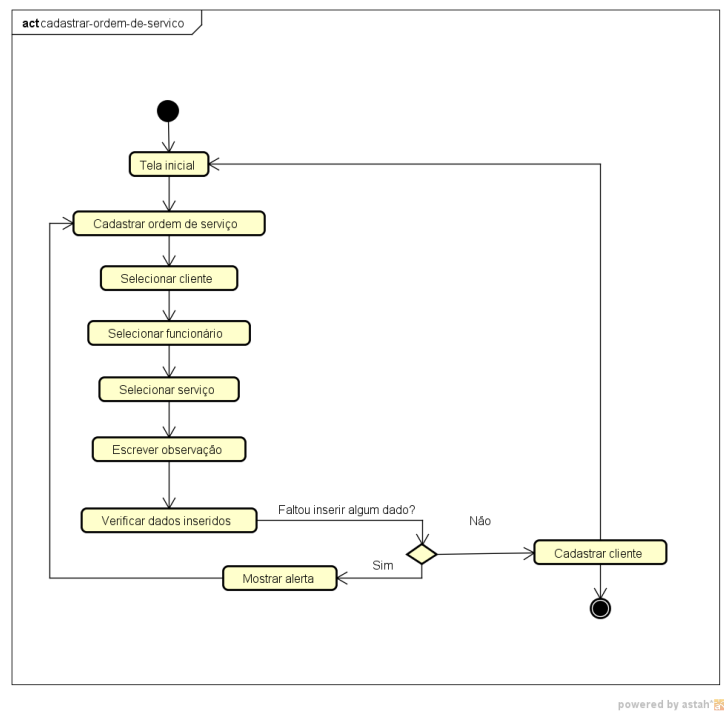
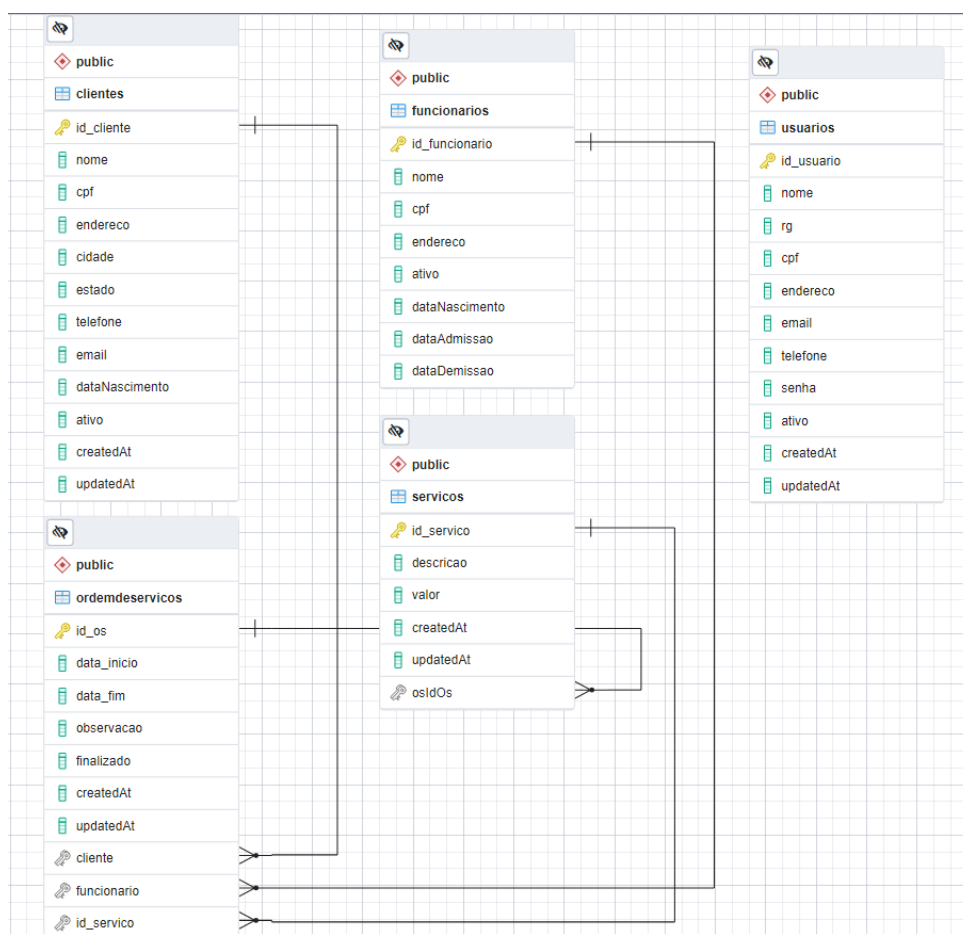


Figura 13 – Diagrama de Atividades - Cadastro do Sistema



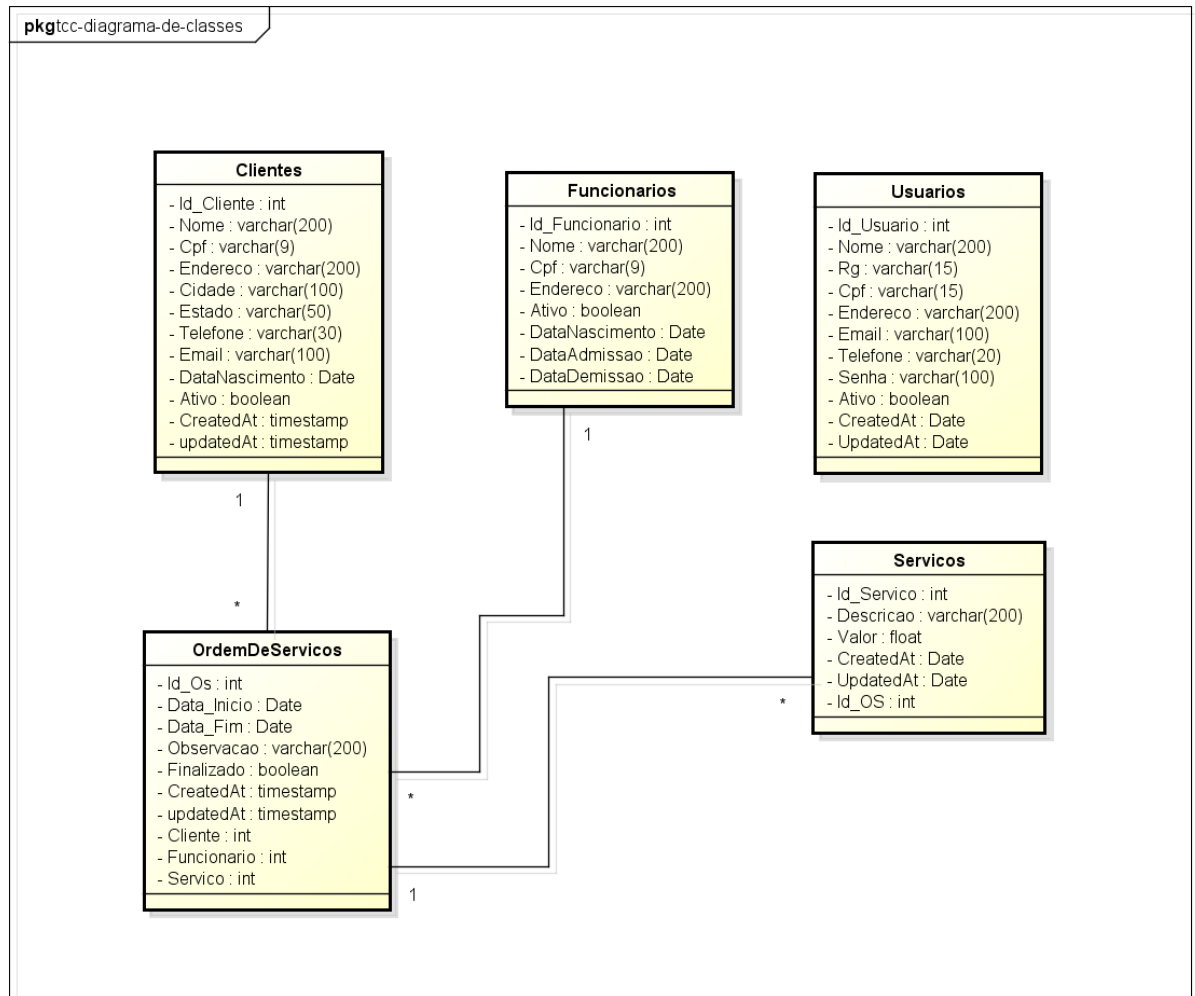
3.3 DIAGRAMA DE ENTIDADE RELACIONAMENTO

Figura 14 – Diagrama de Entidade Relacionamento



3.4 DIAGRAMA DE CLASSES

Figura 15 – Diagrama de Classes



powered by astah

3.5 DICIONÁRIO DE DADOS

O dicionário de dados é um documento que descreve, de forma estruturada, o significado, origem, relacionamento e uso dos dados

3.5.1 ENTIDADE DE CLIENTES

Tabela:	Clientes			
Descrição:	Armazena as informações dos clientes			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de domínio
id_cliente	Código de identificação da tabela	Int		PK / Not Null
nome	Nome do cliente	Varchar	200	Not Null
cpf	CPF do cliente	Varchar	9	Not Null
endereco	Endereço do cliente	Varchar	200	Not Null
cidade	Cidade do cliente	Varchar	100	Not Null
estado	Estado do cliente	Varchar	50	Not Null
telefone	Telefone do cliente	Varchar	30	Not Null
email	Email do cliente	Varchar	100	Not Null
dataNascimento	Data de nascimento do cliente	Date		Not Null
ativo	Identifica se o cliente esta ativo ou não	Boolean		Not Null
createdAt	Identifica quando o cliente foi criado no sistema	Timestamp		Not Null
updatedAt	Identifica a ultima vez que os dados do cliente sofreram alteração	Timestamp		Not Null

3.5.2 ENTIDADE DE FUNCIONÁRIOS

Tabela:	Funcionários			
Descrição:	Armazena as informações dos funcionários			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de domínio
id_cliente	Código de identificação da tabela	Int		PK / Not Null
nome	Nome do funcionário	Varchar	200	Not Null
cpf	CPF do funcionário	Varchar	9	Not Null
endereco	Endereço do funcionário	Varchar	200	Not Null
ativo	Identifica se o funcionário esta ativo ou não	Boolean		Not Null
dataNascimento	Data de nascimento do funcionário	Date		Not Null
dataAdmissao	Identifica quando o funcionário foi admitido na empresa	Date		Not Null
dataDemissao	Identifica quando o funcionário foi demitido da empresa	Timestamp		Not Null

3.5.3 ENTIDADE DE ORDEM DE SERVIÇO

Tabela:	Ordem de Serviços			
Descrição:	Armazena as informações das ordens de serviços			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de domínio
id_os	Código de identificação da tabela	Int		PK / Not Null
data_inicio	Data quando a OS foi inserida no sistema	Date		Not Null
data_fim	Data quando a OS foi finalizada	Date		
observacao	Observação sobre como deve ser realizada a OS	Varchar	200	Not Null
finalizado	Identifica se a OS foi finalizada ou não	Boolean		
cliente	Código do cliente	Int		FK de id_cliente
funcionario	Código do funcionário	Int		FK de id_funcionario
id_servico	Código do serviço	Int		FK de id_servico
createdAt	Data de quando a OS foi criada	Date		
updatedAt	Data quando a OS sofreu alteração	Date		

3.5.4 ENTIDADE DE SERVIÇOS

Tabela:	Serviços			
Descrição:	Armazena as informações dos tipos de serviços			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de domínio
id_servico	Código de identificação da tabela	Int		PK / Not Null
descricao	Descrição do serviço prestado	Varchar	200	Not Null
valor	Valor cobrado no serviço	Float		
osIdOs	Relacionamento entre ordem de serviço e o próprio serviço	Int		FK de id_os
createdAt	Data de quando a OS foi criada	Timestamp		
updatedAt	Data quando a OS sofreu alteração	Timestamp		

3.5.5 ENTIDADE DE ORDEM USUÁRIOS

Tabela:	Usuarios			
Descrição:	Armazena as informações dos usuários			
Campos				
Nome	Descrição	Tipo de dado	Tamanho	Restrições de domínio
id_usuario	Código de identificação da tabela	Int		PK / Not Null
nome	Descrição do serviço prestado	Varchar	200	Not Null
rg	Valor cobrado no serviço	Float		
cpf	Relacionamento entre ordem de serviço e o próprio serviço	Int		FK de id_os
endereco	Endereço do usuário	Varchar	200	Not Null
email	Email do usuário	Varchar	100	Not Null
telefone	Telefone do usuário	Varchar	20	Not Null
senha	Senha do usuário	Varchar	100	Not Null
ativo	Identifica se o usuário está ativo ou não	Boolean		Not Null
createdAt	Data de quando a OS foi criada	Timestamp		
updatedAt	Data quando a OS sofreu alteração	Timestamp		

3.6 TELAS DO SISTEMA

3.6.1 TELA DE LOGIN

A figura a seguir é a primeira tela ao abrir o aplicativo, nela contém o logo da empresa, os campos para inserir o email e senha do usuário e logo abaixo tem os botões para entrar no sistema e cadastrar um novo usuário.

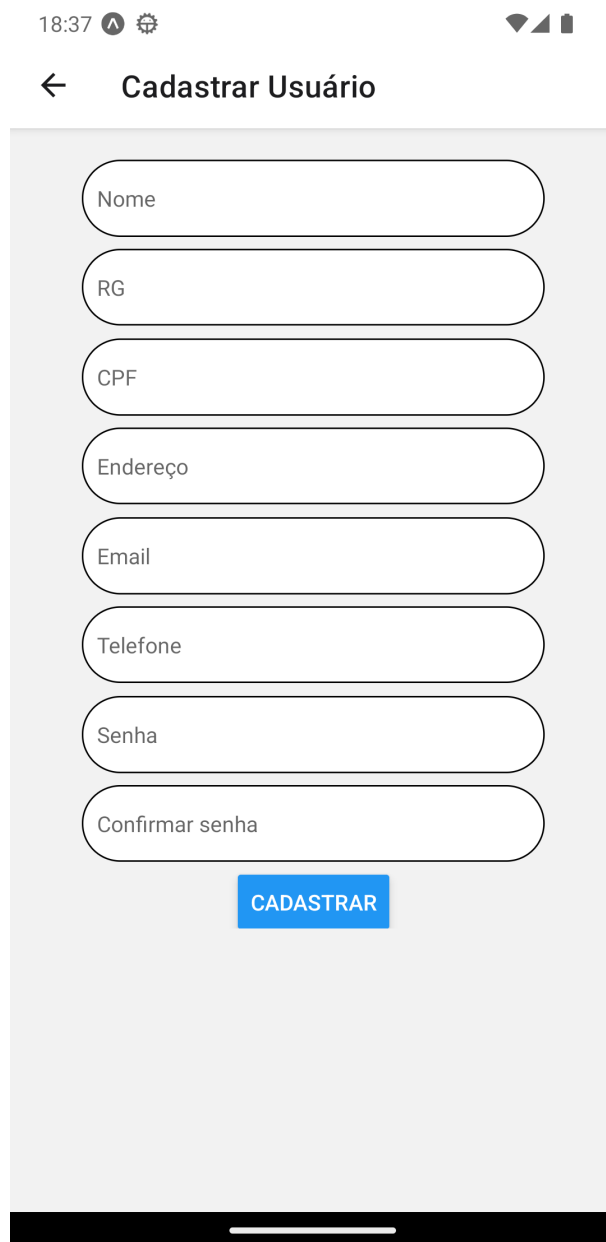
Figura 16 – Tela de Login



3.6.2 TELA CADASTRAR USUÁRIOS

A figura a seguir é por onde é realizado o cadastro de um usuário, cadastro esse que será utilizado na tela anterior para realizar o login no sistema.

Figura 17 – Cadastrar USUÁRIOS

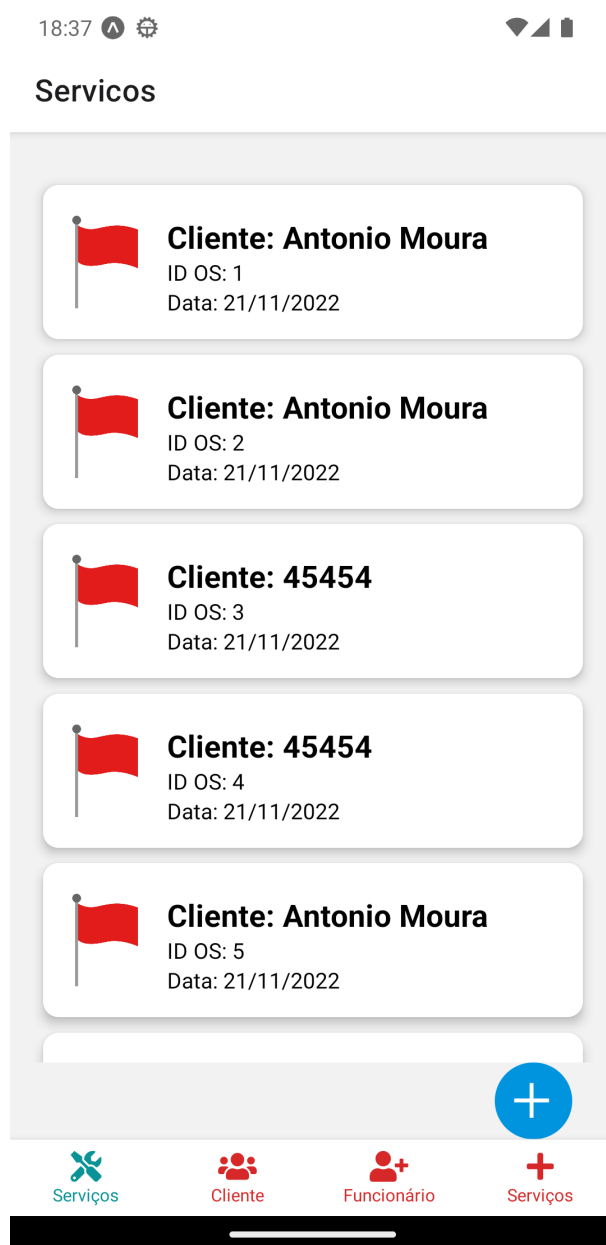


A imagem mostra a interface de um aplicativo móvel para o cadastro de usuários. No topo, há uma barra de status com o horário 18:37, ícones de notificação e configurações, e ícones de conexão de rede e bateria. Abaixo, uma barra de navegação contém um ícone de seta para trás e o título "Cadastrar Usuário". O formulário principal, sobre um fundo cinza claro, possui sete campos de entrada com bordas arredondadas e rótulos internos: "Nome", "RG", "CPF", "Endereço", "Email", "Telefone" e "Senha". Abaixo dos campos, há um botão azul com o texto "CADASTRAR" em branco. Na base da tela, há uma barra preta com uma linha horizontal branca no centro, representando o botão de home.

3.6.3 LISTAGEM DE ORDEM DE SERVIÇO

A figura a seguir é a tela inicial após o usuário realizar o login no sistema, nela contém a listagem das ordens de serviços cadastradas, um botão para cadastrar uma nova OS e no rodapé contém a navegação para outras telas.

Figura 18 – Listagem de OS



3.6.4 TELA CADASTRAR ORDEM DE SERVIÇO

A figura a seguir contém a tela para realizar de fato o cadastro da ordem de serviço.

Figura 19 – Cadastrar ORDEM DE SERVIÇO

A imagem mostra a interface de um aplicativo móvel para criar uma ordem de serviço. No topo, há uma barra de status com o horário 18:37, ícones de notificação e configurações, e ícones de Wi-Fi, sinal de celular e bateria. Abaixo, uma barra de navegação contém um ícone de seta para trás e o título 'Criar Ordem de Serviço'. O formulário principal possui quatro campos: 'Cliente' com o placeholder 'Selecione um cliente', 'Funcionário' com 'Selecione um funcionário', 'Serviços' com 'Selecione um serviço', e 'Observação' com um campo de texto vazio. Um botão azul com o texto 'CADASTAR' está posicionado abaixo dos campos. A base da tela é preta com uma linha branca horizontal no centro.

3.6.5 TELA CADASTRAR CLIENTES

A figura a seguir contém a tela para realizar o cadastro de um cliente, informação essa que será utilizada para criar a OS.

Figura 20 – Cadastrar Clientes

A imagem mostra a interface de um aplicativo móvel para cadastrar clientes. No topo, há uma barra de status com o horário 18:37, ícones de notificação e configurações, e ícones de Wi-Fi e bateria. Abaixo, o título 'Cadastrar Clientes' está em negrito. O formulário contém sete campos de entrada com bordas arredondadas e o texto 'Digite seu' seguido do campo a ser preenchido: nome, CPF, Endereço, Cidade, Estado, Telefone e Email. O campo 'Data de Nascimento' também está presente. Abaixo dos campos, há um botão azul com o texto 'CADASTRAR'. Na base da tela, há uma barra de navegação com quatro ícones: uma chave inglesa (Serviços), duas pessoas (Cliente), uma pessoa com um sinal de mais (Funcionário) e um sinal de mais (Serviços). O ícone 'Cliente' está destacado em verde.

3.6.6 TELA CADASTRAR FUNCIONÁRIO

A figura a seguir contém a tela para realizar o cadastro de um funcionário, informação essa que será utilizada para criar a OS e que deve identificar qual funcionário realizou a ordem de serviço.






Figura 21 – Cadastrar Funcionário

A imagem é uma captura de tela de um aplicativo móvel. No topo, há uma barra de status com o horário 18:37, ícones de notificação e bateria, e ícones de sinal de rede e Wi-Fi. Abaixo, o título da tela é "Cadastrar Funcionário". O formulário principal contém seis campos de entrada de texto, cada um com um rótulo: "Nome", "CPF", "Endereço", "Data de Nascimento", "Data de Admissão" e "Data de Demissão". Abaixo dos campos, há um botão azul com o texto "CADASTRAR". Na base da tela, há uma barra de navegação com quatro ícones e rótulos: "Serviços" (ícone de chave de fenda), "Cliente" (ícone de grupo de pessoas), "Funcionário" (ícone de pessoa com sinal de mais) e "Serviços" (ícone de sinal de mais). O ícone "Funcionário" está destacado em verde.

3.6.7 TELA CADASTRAR TIPOS DE SERVIÇOS


A figura a seguir é a tela onde se cadastrar o serviço prestado pela empresa e que será utilizada para criação da OS.


Figura 22 – Cadastrar Tipos de Serviços


18:37     


Cadastrar Serviços

CADASTRAR

 Serviços

 Cliente

 Funcionário

 Serviços

3.7 LEVANTAMENTO DE INFORMAÇÕES

Este trabalho inclui o desenvolvimento de um sistema de gestão das ordens de serviço da empresa. O sistema não possui módulo financeiro, pois foi decidido neste momento utilizar o sistema apenas para os processos internos da empresa. O sistema possui algumas funções importantes para atender ordens de serviço, como consultar clientes, serviços, consultar histórico de fluxo de caixa (SEBRAE, 2013) e relatórios. Com este sistema, o processo será mais flexível e preciso. Por exemplo, citando o processo de consulta de uma ordem de serviço a pedido do cliente:

a) com o sistema anterior o atendente pesquisava manualmente no arquivo até encontrar o formulário do cliente, o que geralmente demandava de muito tempo pois existem vários formulários no arquivo;

b) com o sistema desenvolvido o atendente pode efetuar a pesquisa no sistema, ou pelo número da ordem de serviço ou pelo nome do cliente, e o sistema retorna as informações da ordem de serviço.

O sistema permite publicar vários relatórios para gestão, nos quais podem ser destacados relatórios, ordens de serviço abertas, ordens de serviço fechadas, ordens de serviço do cliente. Com esses relatórios, as empresas podem gerenciar adequadamente as ordens de serviço e tomar decisões quando necessário. O sistema anterior não tinha nenhum tipo de organização de relatório, tudo o que existia era o arquivamento de um formulário impresso. É difícil criar um relatório a partir deste arquivo, pois tem que ser feito manualmente, registrando as ordens de serviço uma a uma. Portanto, é difícil tomar uma decisão porque a informação não está prontamente disponível.

Outra dificuldade do sistema é calcular o orçamento da ordem de serviço. Após o técnico passar o formulário para o atendente, o orçamento da ordem de serviço deve ser calculado manualmente. Para isso, o atendente deve consultar o catálogo impresso para verificar os preços das peças e serviços, que são calculados automaticamente caso o sistema tenha sido desenvolvido e as peças e serviços tiverem sido cadastrados corretamente no sistema.

3.8 LEVANTAMENTO DE REQUISITOS

Este trabalho consiste no desenvolvimento de um sistema mobile, para efetuar o gerenciamento das ordens de serviço da empresa. O sistema não possui um módulo fiscal, pois neste momento decidiu-se somente a utilização do sistema para processos internos da empresa. O sistema possui quatro funcionalidades principais para atender a ordem de serviço, o cliente, os serviços e os relatórios.

3.8.1 ENTREVISTA REALIZADA NA EMPRESA

a) Qual é o ramo da Empresa?

R: A empresa atua na coleta, remoção e transporte entulho, gestão de estações de transferência de lixo, coleta de entulhos e refugos de obras e demolições, retirada de entulhos após o término das obras, serviços de limpeza urbana.

b) Quais são as pessoas que fazem parte da empresa Rexton?

R: Somente uma pessoa, o proprietário (Tiago)

c) Qual o problema gerado pela falta de um Sistema de Ordem de Serviço?

R: Devido à falta de um software de controle de Ordem de Serviço, não se tem controle dos serviços e reparos executados nos equipamentos da empresa.

d) Qual seria o resultado esperado após a implantação do Sistema de Ordem de Serviço?

R: Com a implantação do Sistema de Ordem de Serviço espera-se que se tenha agilidade, praticidade, maior controle dos materiais gastos, e das atividades rotineiras realizadas na empresa e no arquivamento dos dados de uma forma segura e de fácil acesso.

4 OBJETIVOS

O objetivo deste trabalho é apresentar o desenvolvimento de um sistema mobile para gerar relatórios de clientes bem como gerenciar as ordens de serviços da empresa Rexton.

4.1 ESPECÍFICOS

Os objetivos específicos do trabalho proposto são:

- a) Gerenciar de forma eficaz a prestação de serviços através do controle de ordens de serviço;
- b) Gerir informações dos clientes;
- c) Emissão de relatórios contendo entradas e saídas do caixa;
- d) disponibilizar acesso aos históricos dos clientes e a relatórios das ordens de serviço e dos serviços prestados.

5 METODOLOGIA

O sistema foi desenvolvido com tecnologia open-source [Nascimento \(2014\)](#), com linguagem de programação Javascript e utilizando a biblioteca de desenvolvimento React-Native [Junior \(2020\)](#) juntamente com o banco de dados relacional PostgreSQL [Souza \(2020\)](#). Para o auxílio da modelagem do sistema, será utilizada as técnicas de UML (Linguagem de Modelagem Unificada). Por se tratar de um sistema mobile ele estará disponível em qualquer lugar e a qualquer hora, desde que exista um smartphone com conexão a internet. Além disto, não serão necessários altos investimentos em tecnologias, pois o sistema irá rodar pelo diretamente no celular.

5.1 REACT-NATIVE

Como afirma CABRAL, “React Native [Junior \(2020\)](#) é um framework de desenvolvimento. Desenvolvido por engenheiros do Facebook, contém uma coleção de ferramentas para criar aplicativos móveis nativos para plataformas iOS e Android utilizando o mais moderno desenvolvimento front-end.”

React Native é usado para o desenvolvimento da parte móvel de aplicativos de trabalho e é um framework muito rico que tem muito poder quando se trata de construir aplicativos.

5.2 NODE.JS

Segundo Lopes, o Node.JS ([CAMPOS et al., 2016](#)) é uma plataforma de desenvolvimento de aplicações server-side baseada no uso do interpretador do motor JavaScript V8 do Google, que utiliza apenas código JavaScript, o mesmo utilizado pelo navegador Google Chrome.

Moreira [Campos et al. \(2016\)](#) entende que o principal objetivo do Node.js é fornecer uma maneira fácil de construir programas de rede escaláveis. Ao invés de criar uma thread de conexão, cada requisição aciona um evento que é executado dentro da engine, evitando deadlocks e desabilitando locks. Suportando assim dezenas de milhares de conexões simultâneas.

O backend da aplicação é feito com Node.js, o que é muito útil para o desenvolvimento. Não é difícil de aprender. Existem muitas bibliotecas que ajudam no processo de desenvolvimento.

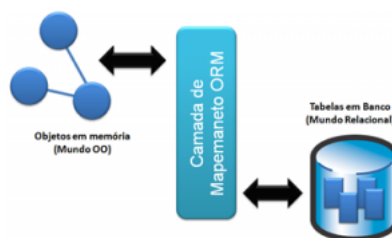
5.3 POSTGRESQL

O PostgreSQL, comumente conhecido como Postgres, é um dos cinco SGBDs relacionais mais usados no mercado. De código aberto e gratuito também estão entre as primeiras opções que os desenvolvedores consideram ao construir um projeto. O PostgreSQL é um sistema de banco de dados relacional de código aberto que existe há mais de 30 anos e tem uma sólida reputação de confiabilidade e robustez. (SOUZA, 2020)

Algumas características chave do banco de dados PostgreSQL que o tornam único e amplamente favorecido quando comparado a outros bancos de dados. Atualmente, é o segundo banco de dados mais utilizado, ficando atrás apenas do MySQL. De acordo com Barros (2016) o PostgreSQL oferece verdadeira semântica ACID para transações e tem total suporte para chaves estrangeiras, joinins, views, triggers e procedimentos armazenados, em muitos idiomas diferentes. Ele inclui a maioria dos tipos de dados SQL como INTEGER, VARCHAR, TIMESTAMP, e BOOLEAN. Também suporta o armazenamento de grandes objetos binários, incluindo imagens, vídeos ou sons. Ele é confiável, pois tem uma grande rede de suporte integrada à comunidade. O PostgreSQL é um banco de dados tolerante a falhas graças ao seu registro write-ahead.

5.4 TYPEORM

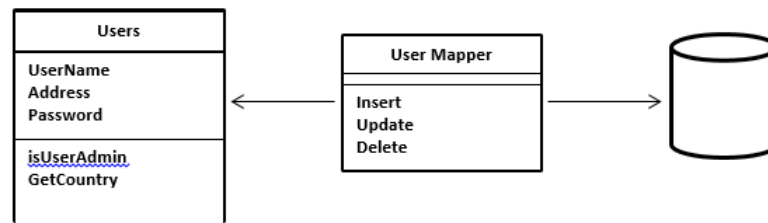
ORM (Object Relational Mapper) é uma técnica de mapeamento objeto relacional que permite fazer uma relação dos objetos com os dados que os mesmos representam. Este crescimento tem se dado principalmente pelo fato de muitos desenvolvedores não se sentirem a vontade em escrever código SQL e pela produtividade que esta técnica nos proporciona. Existem ótimos ORM's como Hibernate, NHibernate, Entity Framework e o TypeORM.



Existem dois mundos: o relacional e o orientado a objetos, no mundo relacional prevalecem princípios matemáticos com a finalidade de armazenar e gerenciar corretamente os dados, de forma segura e se trabalha com a linguagem SQL que é utilizada para dizer o banco de dados “O QUE?” fazer e não como fazer.

Já no mundo orientado a objetos trabalhamos com classes e métodos, ou seja, se trabalha com fundamentados na engenharia de software e seus princípios que dizem

“COMO” fazer. O ORM é justamente, a ponte entre estes dois mundos, ou seja, é ele quem vai permitir que a aplicação armazene os objetos no banco de dados. Para isto se faz necessário fazer um mapeamento dos seus objetos para as tabelas do banco de dados.



A figura anterior traz uma ideia de como o ORM trabalha. Ele faz o mapeamento da sua classe para o banco de dados e cada ORM tem suas particularidades para gerar o SQL referente a inserção do objeto que corresponde a uma tabela no banco de dados e realizar a operação. Utilizando um ORM, também se ganha produtividade, pois deixa-se de escrever os comando SQL para deixar que o próprio ORM, faça isto por você. ([TYPEORM, 2011](#))

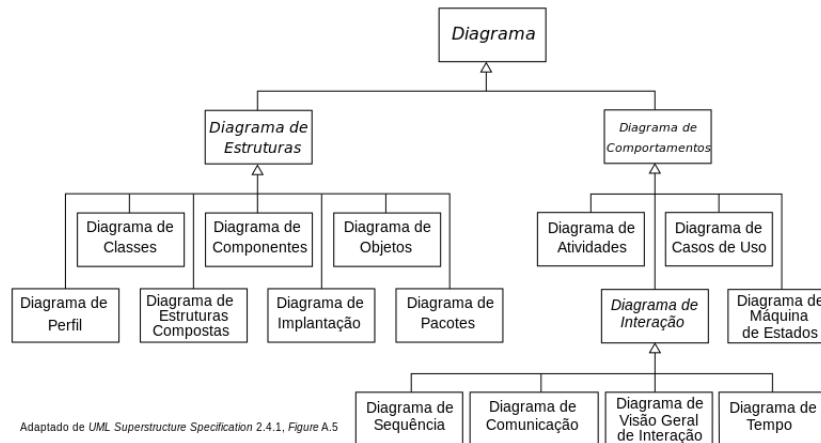
5.5 ASTAH

O software foi desenvolvido no Japão na plataforma Java, o que garante sua portabilidade para qualquer plataforma que possui JVM (Máquina Virtual Java). JUDE (Ambiente para Desenvolvedores UML e Java) obteve o prêmio “Produto de Software do Ano 2006”, pela Agência de Promoção de Informação Tecnológica no Japão. Anteriormente conhecido como JUDE, ele funciona nas plataformas Windows, Mac e Linux. ([ASTAH-UML, 2017](#))

Modelagem define os seus sistemas de uma forma que é mais fácil de entender, simples de comunicar e mais em contato com as pessoas que irão utilizar. Na área de Engenharia de Software, a UML (Linguagem de Modelagem Unificada) é uma linguagem de modelagem que permite representar um sistema de forma padronizada. Astah utilizada nos diagramas dinâmicos, essa ferramenta já é bastante consolidada, voltada para a modelagem de sistemas utilizando a UML, utiliza como recurso adicional a modelagem MAS ML (Modelagem de um Sistema Multiagente).

UML, criada por Grady Booch, Ivar Jacobson & Jaimes Rumbaugh. É hoje o método mais comum para o paradigma orientado a objetos. Os objetivos da UML são: especificação, documentação, estruturação para sub-visualização e maior visualização lógica do desenvolvimento completo de um sistema de informação. UML 2.2, conforme a OMG (Object Management Group – organização internacional que aprova padrões abertos para aplicações orientadas a objetos), possui 14 tipos de diagramas, divididos em duas

grandes categorias: Estruturais e Comportamentais. Sete tipos de diagramas representam informações estruturais, e os outros sete representam tipos gerais de comportamento, incluindo quatro em uma sub-categoria que representam diferentes aspectos de interação. Estes diagramas podem ser visualizados de forma hierárquica, como apresentado no padrão de diagrama de classes a seguir.



No Astah é possível modelar 9 dos 14 tipos de diagramas que são eles:

* Diagrama de Estruturas: Diagrama de Classes, Diagrama de Estruturas Compostas, Diagrama de Componentes e Diagrama de Implantação. * Diagrama de Comportamentos: Diagrama de Atividades, Diagrama de Interação, Diagrama de Casos de Uso e Diagrama de Máquina de Estados. * Diagrama de Interação : Diagrama de Sequência e Diagrama de Comunicação.

5.6 VISUAL STUDIO CODE

Em 2015, a Microsoft introduziu um editor de código para desenvolver aplicativos da Web chamado Visual Studio Code, ou [VSCODE \(2016\)](#), abreviado. Anunciado durante o Build, um evento anual para desenvolvedores nos EUA, é uma ferramenta leve e multiplataforma disponível para Windows, Mac OS e Linux e atende a uma ampla variedade de projetos, não apenas ASP. .NET e Node.js. Além disso, o editor suporta a sintaxe de Python, Ruby, C++ e outras linguagens.

Além de ser totalmente gratuito, na segunda metade do lançamento, durante o evento Connect(), o editor foi anunciado como open source, e o código está disponível no GitHub, o que permite que a comunidade técnica desenvolva e contribua com seu desenvolvimento e facilitar a criação de extensões e novas funcionalidades.

O VS Code será utilizado em todo o processo de desenvolvimento, tanto na parte do front-end quanto na parte [Back-end \(2021\)](#), e é uma ferramenta muito importante para entrega de aplicativos.

6 CONCLUSÃO

Por meio deste trabalho foi possível identificar as vantagens da substituição de sistemas utilizados manualmente por informatizados a fim de atingir seus objetivos com maior precisão, agilidade e praticidade. Uma proposta de monografia foi concluída com sucesso com o objetivo de desenvolver um sistema de ordens de serviço para facilitar o processo de abertura de solicitações de serviço por meio de uma aplicação android.

Como resultado do trabalho, foi criado um aplicativo para Android para que os usuários recebam informações sobre as ordens de serviço e para a comunicação entre ambos, foi criado um Web Service baseado na arquitetura REST.

Após a instalação e utilização do sistema no dia a dia, espera-se reduzir erros ocorridos antes da utilização desta nova ferramenta, como falta de controle sobre retirada de material, agendamento de serviços, controle de clientes, atendimento e reajustes de preços de serviços prestados.

REFERÊNCIAS

- ASTAH-UML. *Tutorial da ferramenta de modelagem ASTAH*. 2017. Publisher: uaberta. Disponível em: <<https://www.uaberta.unisul.br/sgc/downloadArquivoConteudo.processa?ead=1.502323005503715E121520391051183&arquivoId=41148&comunidadeId=44>>. Acesso em: 24 outubro. 2022. Citado na página 41.
- Back-end. *Front-end e Back-end: entenda essas duas áreas do desenvolvimento*. 2021. Disponível em: <<https://blog.somostera.com/carreiras-digitais/front-end-e-back-end>>. Acesso em: 09 Maio. 2022. Citado na página 42.
- BARROS, P. *O que é ACID?* 2016. Disponível em: <<https://medium.com/opensanca/o-que-%C3%A9-acid-59b11a81e2c6>>. Citado na página 40.
- CAMPOS, D. P. G. et al. Aplicação de armazenamento em nuvem utilizando a plataforma node.js. *Revista Eletrônica Científica de Ciência da Computacao*, v. 11, n. 1, p. 13, nov 2016. Citado na página 39.
- JUNIOR, A. C. *React Native: o que é, suas funcionalidades e suas vantagens*. 2020. Disponível em: <<https://ateliware.com/blog/react-native>>. Acesso em: 01 Maio. 2022. Citado na página 39.
- NASCIMENTO, A. *O que é open source?* 2014. Publisher: CanalTech. Disponível em: <<https://canaltech.com.br/produtos/O-que-e-open-source/>>. Acesso em: 04 Maio. 2022. Citado na página 39.
- NITAHARA, A. *IBGE: setor de serviços tem crescimento de 2,6% em julho*. 2020. Disponível em: <<https://agenciabrasil.ebc.com.br/economia/noticia/2020-09/ibge-setor-de-servicos-tem-crescimento-de-26-em-julho>>. Citado na página 11.
- REZENDE, D.; ABREU, A. de. *Tecnologia da informação aplicada a sistemas de informação empresariais: o papel estratégico da informação e dos sistemas de informação nas empresas*. Atlas, 2008. v. 9. ISBN 9788522450039. Disponível em: <<https://books.google.com.br/books?id=-sduPgAACAAJ>>. Citado na página 11.
- SEBRAE. *Fluxo-de-Caixa*. 2013. Disponível em: <<https://www.sebrae.com.br/sites/PortalSebrae/artigos/fluxo-de-caixa-o-que-e-e-como-implantar,b29e438af1c92410VgnVCM100000b272010aRCRD>>. Acesso em: 29 Abril. 2022. Citado na página 36.
- SENAC. *SERVIÇO NACIONAL DE APRENDIZAGEM COMERCIAL*. 2001. OCLC: 986757411. Citado na página 11.
- SILVA, D. da. *Ordem de serviço é importante? Guia completo e definitivo!* 2020. Disponível em: <<https://www.zendesk.com.br/blog/ordem-de-servico/>>. Acesso em: 25 Abril. 2022. Citado na página 11.
- SOUZA, I. de. *PostgreSQL: saiba o que é, para que serve e como instalar*. 2020. Disponível em: <<https://rockcontent.com/br/blog/postgresql/>>. Acesso em: 04 Maio. 2022. Citado 2 vezes nas páginas 39 e 40.

TYPEORM, D. *Conceitos Básico em ORM (Object Relational Mapper)*. 2011. Disponível em: <<https://www.devmedia.com.br/orm-object-relational-mapper/19056>>. Acesso em: 24 outubro. 2022. Citado na página 41.

VSCODE. 2016. Disponível em: <<https://www.devmedia.com.br/introducao-ao-visual-studio-code/34418>>. Acesso em: 09 Maio. 2022. Citado na página 42.